

095135



095135

706186



AUDITING COMPUTERS WITH A TEST DECK

(With Emphasis On
Payroll Applications)

This is a booklet on test decks
and how they can be used to
evaluate controls in computer
systems.

BY THE COMPTROLLER GENERAL
OF THE UNITED STATES

1975

706186

FOREWORD

The computer age has brought many new challenges for auditors. The speed and complexity of computers, combined with the lack of human readable data in many phases of computer processing, have necessitated many changes in auditing techniques. This booklet describes one such technique, called "test decking." Although test decks have been in existence for years, we believe a restatement of the basic concepts, along with an explanation of our recent experience on automated payroll applications, will be helpful to the auditor who must work in the computer environment.

A test deck is a set of simulated transactions which can be processed through a computer system to see whether proper transactions will be processed accurately and improper ones identified and rejected. This technique helps to determine whether the computer programs process data accurately and whether the controls in the system can prevent improper results, such as exorbitant payroll checks.

However, using a test deck should not be regarded as a complete audit of a computer system.

For example, it will not disclose (1) invalid but properly prepared input data or (2) some changes in computer programs that might be made to produce fraudulent results. It must be supplemented with other procedures to provide satisfactory audit coverage. When used correctly, however, a test deck does provide a great deal of information about the way computer programs work.

This booklet provides auditors with general information about the test deck technique and some specific guidance for using it to audit Federal agency computer systems. The booklet is based primarily on recent experience in applying test decks to batch-processed payroll systems. However, the basic concepts and general approaches discussed can be applied to developing and using test decks for testing programmed controls and procedures in any administrative or accounting data processing system which records, manipulates, and summarizes data.

Thomas B. Steels

Comptroller General
of the United States

December 1975

DESCRIPTION AND

ADVANTAGES AND D

Advantages

Disadvantage

DESIGN AND USE O

Test deck de

Tests of

transa

Tests us

Test to

establ

Entering tes

Analyzing pr

APPLYING TEST DE

UPDATE MASTER

Using copied

to set up

C o n t e n t s

	<u>Page</u>
DESCRIPTION AND PURPOSE OF TEST DECKS	1
ADVANTAGES AND DISADVANTAGES OF TEST DECKS	4
Advantages	4
Disadvantages	5
DESIGN AND USE OF TEST DECKS	8
Test deck design	8
Tests of normally occurring transactions	11
Tests using invalid data	11
Test to violate established edit checks	14
Entering test data	14
Analyzing processing results	16
APPLYING TEST DECKS AGAINST PROGRAMS THAT UPDATE MASTER RECORDS	18
Using copied and/or simulated master records to set up a separate test file	18

	<u>Page</u>	<u>DESCRIP</u>
Using special audit records maintained in the current master file	20	A test deck designed to test of programmed code computer system both expected and happen during audit Before the transaction results are "predicted tion that the code effective intervention specified) so that with the predicted
Conclusion	22	
SUMMARY OF SUGGESTED STEPS IN DESIGNING AND USING A TEST DECK	23	
APPENDIX I--HOW TEST DECKS WERE DESIGNED AND USED IN TWO REVIEWS OF AUTOMATED PAYROLL SYSTEMS	26	
APPENDIX II--TYPICAL PAYROLL TRANSACTIONS TO BE INCLUDED IN A TEST DECK	30	

Test decks
 items test decks,
 including both
 (2) program test
 computer programs that
 are processed and
 (In reviewing batches
 items (see app. I
 (see app. II) to
 accurately process

DESCRIPTION AND PURPOSE OF TEST DECKS

A test deck is a set of simulated transactions designed to test for the existence and effectiveness of programed controls and procedural operations in computer systems. The transactions should represent both expected and conceivable conditions that could happen during actual data processing operations. Before the transactions are run, expected test results are "predetermined" (calculated on the assumption that the computer program being tested contains effective internal controls and will perform as specified) so that actual results can be compared with the predetermined results.

Test decks are of two general types: (1) systems test decks, which test all types of conditions, including both manual and computer procedures, and (2) program test decks, which test the logic in computer programs to determine how various transactions are processed and how certain controls function. (In reviewing batch-processed automated payroll systems (see app. I), GAO is using program test decks (see app. II) to find out whether computer programs accurately process pay and leave data.)

A test deck is useful for making compliance tests of a computer system's internal controls,¹ particularly when (1) much of the internal control system is embodied in computer programs and/or (2) gaps in visible audit trails make it difficult or impractical to trace input to output or to verify calculations.

¹Test decks do not test the accuracy of file data or transactions. To make substantive tests of data and transactions, the auditor must perform manual tests or use a package of computer program routines especially designed for audit purposes.



*"Will the computer
exorbitant amount (for \$50,000)?" Wit*



"Will the computer print out checks to a nonexistent person for an exorbitant amount (say a biweekly payroll check to Donald Duck for \$50,000)?" With a test deck you can find out.

ADVANTAGES AND DISADVANTAGES OF TEST DECKS

The auditor should consider the following advantages and disadvantages of test decks from the standpoint of audit objectives and other relevant factors.

ADVANTAGES

1. A test deck can be designed for any program, system, or equipment.
2. Test results can be readily checked.
3. The test data processes either correctly or incorrectly.
4. The auditor can choose the types and combinations of transactions or procedures to be tested.
5. The auditor can accept test results without actually tracing data through processing stages.

6. Extens:
system:
program
working
and out
procedu

DISADVANTAGES

1. A test applica designe
2. It test in time incorpo would a as chan rules, c
3. It is co a good t

6. Extensive technical knowledge of computer systems or the ability to write computer programs is not required. (However, a working knowledge of file structure, input and output formats, and data processing procedures is necessary.)

DISADVANTAGES

1. A test deck is valid only for the single application or program for which it is designed.
2. It tests procedures only for a given point in time and therefore must be updated to incorporate any system or other changes that would affect the validity of the tests, such as changes in file structure, statutes, rules, or regulations.
3. It is costly and time consuming to develop a good test deck.

4. It is impractical, if not impossible, to design a test deck to test every conceivable situation in a typical administrative or accounting type computer system because of the virtually limitless variety of transactions and conditions that could occur.





Audit staff discusses simulated transactions to be included in test deck.

DESIGN AND USE OF TEST DECKS

TEST DECK DESIGN

To design an adequate test deck, the auditor must be familiar with (1) relevant laws and regulations and include their provisions in the simulated transactions and (2) data processing system procedures and input and output formats for all types of transactions to be processed. Also helpful is a preliminary knowledge of system objectives and operating procedures obtained by reviewing and analyzing system flowcharts, operating instructions, and other documentation. This knowledge can alert the auditor to possible system weaknesses for which unique test transactions should be designed.¹

¹The auditor should not overlook using test decks developed by operating personnel. These test decks, often used for "debugging" data processing programs during the development of computer systems, may fulfill the auditor's requirements with only minor alteration. Before using an agency's test deck, however, the auditor should determine whether it includes tests for all conditions which should be tested, including those involving invalid data. If it does not, the auditor should add simulated transactions designed to test for those conditions. He may also wish to consider using one of various test data packages available commercially.

To be effective, test actions having input data--v operations are controls.

Only one test against each simulated evaluation ensuring that to other test transactions master record.

¹A master record data treated as system, a payroll on an individual status.) Master machine-readable during data processing collectively a payroll system make up the payroll

To be effective, a test deck should use transactions having a wide range of valid and invalid input data--valid data for testing normal processing operations and invalid data for testing programmed controls.

Only one test transaction should be processed against each master record.¹ This permits an isolated evaluation of specific program controls by insuring that test results will not be influenced by other test transactions processed against the same master record.

¹A master record is a collection of related items of data treated as a unit. (For example, in a payroll system, a payroll master record contains information on an individual employee's pay history and status.) Master records are permanent records, in machine-readable form, and are updated periodically during data processing runs. They are referred to collectively as the "master file." Thus, in a payroll system, the master records for all employees make up the payroll master file.



Coding test transactions onto key punch documents

General t
tested are dis

Tests of norma

To test a
ately process
clude transact
ample, in a pa
transactions i
overtime pay,
(such as shift
records for ne
ing master rec

Tests using in

Testing f
programed cont

Examples of te
jected or "fla

1. Enter
chara

General types of conditions which should be tested are discussed below.

Tests of normally occurring transactions

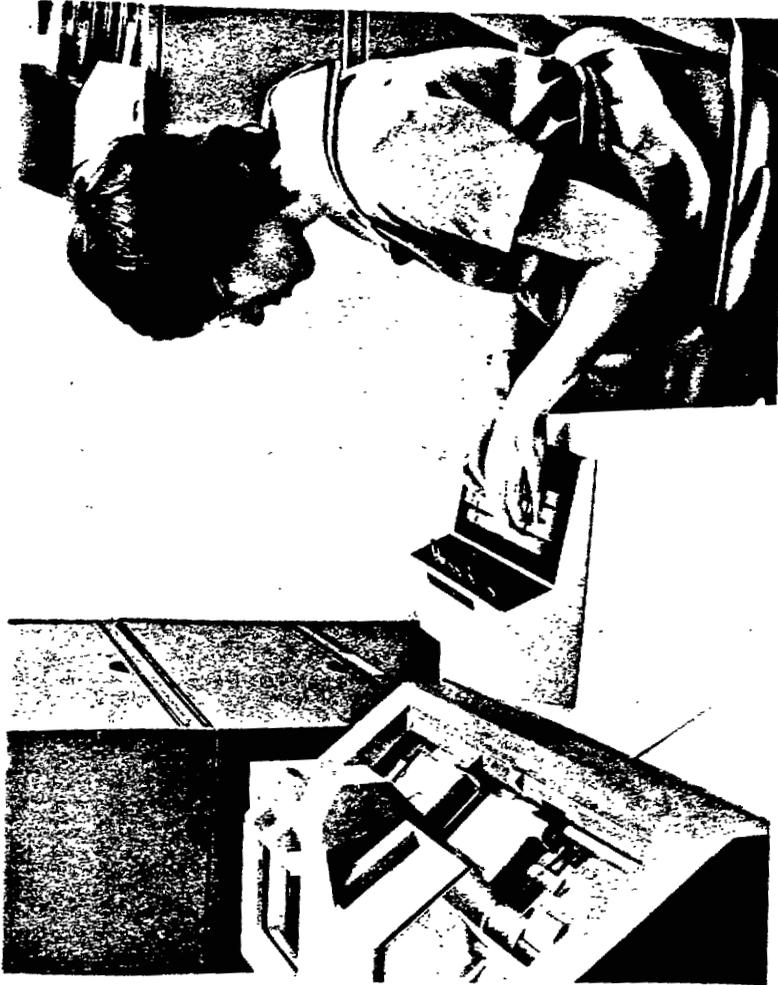
To test a computer system's ability to accurately process valid data, a test deck should include transactions that normally occur. For example, in a payroll system, normally occurring transactions include the calculation of regular pay, overtime pay, and some other type of premium pay (such as shift pay), as well as setting up master records for newly hired employees and updating existing master records for other employees.

Tests using invalid data

Testing for the existence or effectiveness of programed controls requires using invalid data.

Examples of tests for causing invalid data to be rejected or "flagged" are:

1. Entering alphabetic characters when numeric characters are expected and vice versa.



Key punching test transactions

2. Using inv
numbers.
3. Using inc
specific
4. Entering n
tive amoun
5. Entering il
which logic
6. Entering a
does not mat
lished by op
ling tables.
codes for em
system are A
entered woulc
or C. Anothe
amount which
ling salary t.
7. Entering trans
will violate 1

2. Using invalid account or identification numbers.
3. Using incomplete or extraneous data in a specific data field or omitting it entirely.
4. Entering negative amounts when only positive amounts are valid and vice versa.
5. Entering illogical conditions in data fields which logically should be related.
6. Entering a transaction code or amount that does not match the code or amount established by operating procedures or controlling tables. For example, if the valid codes for employee status in a payroll system are A, B, and C, the code to be entered would be something other than A, B, or C. Another example is entering a salary amount which is incompatible with a controlling salary table.
7. Entering transactions or conditions that will violate limits established by law or

by standard operating procedures. An example, in a payroll system, is the entry of $X + 2$ dollars as an employee's gross pay when X dollars is the maximum gross pay allowed by law for the highest grade.

Tests to violate
established edit checks

From system documentation, the auditor should be able to determine what edit routines are included in the computer programs to be tested. He should then create test transactions to violate these edits to see whether they, in fact, exist.

ENTERING TEST DATA

After the types of test transactions have been determined, the test data should be put into correct entry form. If the auditor wishes to test controls over both input and computer processing, he should feed the data into the system on basic-source documents for the agency to convert into machine-readable form (on punched cards, paper tape, magnetic tape, etc.). If he is only testing computer





Auditor submits completed test deck to computer operator for processing.

processing controls, he should enter the data in machine-readable form.

ANALYZING PROCESSING RESULTS

Before processing test data through the computer, the auditor must predetermine the correct result for each test transaction for comparison with actual results. Any difference between actual and predetermined results indicates a weakness in the system's automated controls, which should be compensated for by an alternative control in the manual part of the system. If it is not, the auditor should determine the effect of the weakness on the accuracy of master file data and on the reliability of reports and other computer products.

Even if a weakness in automated controls is compensated for by an alternative manual control, the auditor should consider recommending replacement of the alternative control by an automated control. Controls should be automated whenever practical to increase the speed of data processing, to avoid unnecessary personnel costs, and to reduce errors that commonly occur in manually performed operations.





Auditors analyze the results of test deck processing.

APPLYING TEST DECKS AGAINST PROGRAMS
THAT UPDATE MASTER RECORDS

There are two basic approaches to test programs for master records updating. In one approach, copies of actual master records and/or simulated master records are used to set up a separate master file for the test. In the other, special audit records, kept in the agency's current master file, are used.¹

USING COPIED AND/OR SIMULATED
MASTER RECORDS TO SET UP
A SEPARATE TEST FILE

To use the first approach, the auditor must have a part of the agency's master file copied to

¹ A possible third approach for using master records is to process test transactions against actual master records in the agency's current master file. In this approach, the auditor includes test data with "live" transactions and processes it against actual master records during a regular processing run. Complications inherent in this approach are that (1) tests processed with actual transactions against actual records must be carefully controlled to prevent undesired changes or results and (2) test data used to update actual master records must later be reversed out of the system, which is a time-consuming process requiring the highest order of competence, coordination, and precision. We do not recommend this approach because of these complications and the substantial risk of inadvertent changes and distortions in agency records and reports.

create a test m:
file, he select:
then updates th
valid data by u
the transaction

Master re
preparing sourc
the program the
master file. I
records are the

An advant
they can be ta
they eliminate
agency records
however, when
creation can b
pared to the r
a part of the

Often, th
test master fi:

¹ A computer ca
actual recor

create a test master file. From a printout of this file, he selects records suitable for the test. He then updates the test file with both valid and invalid data by using the agency's programs to process the transactions making up the test deck.

Master records can be simulated¹ most easily by preparing source documents and processing them with the program the agency uses to add new records to its master file. Procedures for test decking simulated records are the same as those for copied records.

An advantage of using simulated records is that they can be tailored for particular conditions and they eliminate the need to locate and copy suitable agency records. This advantage is usually offset, however, when many records are needed because their creation can be complex and time consuming when compared to the relatively simple procedure of copying a part of the agency's master file.

Often, the most practical approach is to use a test master file which is a combination of copied

¹A computer cannot distinguish simulated from actual records.

and simulated master records. In this approach, copied records are used whenever possible and simulated records are used only when necessary to test conditions not found in the copied records.

By using copied and simulated master records in a separate test file, the auditor avoids the complications and dangers of running test data in a regular processing run against an agency's current master file. (These complications and dangers are discussed below.) A disadvantage of copied and simulated records is that computer programs must be loaded and equipment set up and operated for audit purposes only, thus involving additional cost.

USING SPECIAL AUDIT RECORDS
MAINTAINED IN THE CURRENT MASTER FILE

Special records can be kept in the agency's current master file for audit purposes only. Using this approach, the auditor includes test data with "live" transactions and processes it against the master file during a regular processing cycle. The special audit records are easy to identify because

they are given simulated. For ex-
ences to none:
processing, th
"live" activit
centers.

This appo
grams and perf
test data. Si
mal operating
cessed faster
uses copied or
ate test file.

The audit
processing tes
transactions n
agency's maste
sonnel conceiv
example, by ch
ences to real
thermore, this
reports produc
thus necessita

they are given references which show they are simulated. For example, the records may contain references to nonexistent cost centers; during normal processing, these records remain inactive because "live" activity does not affect fictitious cost centers.

This approach makes it unnecessary to load programs and perform other setup work solely to process test data. Since a system can be tested under normal operating procedures, a test deck can be processed faster and at less cost than when the auditor uses copied or simulated master records in a separate test file.

The auditor should recognize, however, that processing test transactions concurrently with "live" transactions may inadvertently change or distort the agency's master records. Also, data processing personnel conceivably could activate these records (for example, by changing fictitious cost center references to real ones) and use them fraudulently. Furthermore, this approach may distort management reports produced by the normal processing cycle, thus necessitating clerical adjustments to correct

the reports for the presence of test data. Finally, data processing personnel may object to having these audit records "clutter" their master file. Therefore, these tests must be carefully controlled.

CONCLUSION

The auditor must justify either (1) the risk of using actual master records or special audit records in the current master file or (2) the cost of having additional runs of computer programs being tested. Processing test data concurrently with "live" data to update current master records--which could be inadvertently altered or destroyed during the test--is not recommended. The use of copied and/or simulated master records in a separate test file, which does not entail testing as a part of a regular processing run, is preferred.

SU

DES

In design
tor should:

1. Deter
incl
2. Dete:
be u:
3. Obta
prep
proc
4. Prin
recc

¹ Assuming the
and/or simu:
master reco

SUMMARY OF SUGGESTED STEPS IN
DESIGNING AND USING A TEST DECK

In designing and using a test deck, the auditor should:

1. Determine the types of transactions to be included in the test deck (see pp. 8 to 17).
2. Determine the types of master records to be used (see pp. 18 to 22).
3. Obtain copies of master records and/or prepare simulated master records for processing with the test transactions.¹
4. Print out the contents of all such master records before the test.

¹Assuming the auditor has decided to use copied and/or simulated master records rather than "live" master records.

5. Predetermine the correct end result for each test transaction for comparison with actual processing results. (Appendix I shows how this was performed in two payroll system reviews.)
6. Verify that the programs used for processing the test transactions are the same as those used for normal payroll processing.
7. Ascertain that any changes to these programs during the review have been documented and that the changes have been tested and approved by the agency.¹
8. Arrange with data processing personnel for processing test transactions and producing output in a useful form.
9. Print out the contents of all master records used in the test and compare with the

prec
whet
pect

10. Dete
mate
mast
(see

¹The auditor should be aware that agency employees can change computer programs by removing fraudulent or irregular routines.

¹The auditor must
tion retrieval
the master rec
identify all c

predetermined results (see #5) to see whether the programs performed as expected.¹

10. Determine what impact weaknesses in automated controls have on the accuracy of master file data and computer output (see p. 16).

¹The auditor may wish to use a computerized information retrieval and analysis routine to compare the master records before and after the test to identify all changes made.

APPENDIX I

HOW TEST DECKS WERE DESIGNED AND USED

IN TWO REVIEWS

OF AUTOMATED PAYROLL SYSTEMS

In making two recent reviews of automated civilian payroll systems, we used test decks to test the agencies' computer programs for processing pay and leave data.

We first reviewed all available documentation for the manual and automated parts of each system. To understand the manual operations, we interviewed payroll supervisors and clerks, reviewed laws and regulations relating to pay and leave, and familiarized ourselves with standard payroll operating procedures. For the automated part of each system, we interviewed system designers and programmers and reviewed system and program documentation and operating procedures.

After acquiring a working knowledge of each system, we decided to test computer programs used to update payroll master records and those used to calculate biweekly pay and leave entitlements. Although we were concerned primarily with these particular programs, we decided that other programs used in the

normal biweekly programs for pay and leave records, be tested to see

We then checked the pay and leave transactions against internal control procedures and regulations and regulations operating procedures and regulations. These transactions were made effective through the agency's payroll system for each master record.

The best master record to use copies with simulated data not found in

According to the agency's pay

APPENDIX I

normal biweekly payroll processing cycle (such as programs for producing pay and leave history reports, leave records, and savings bond reports) should also be tested to see how they would handle test data.

We then designed a test deck of simulated pay and leave transactions to test the effectiveness of internal controls, compliance with applicable laws and regulations, and the adequacy of standard payroll operating procedures. The test deck included transactions made up of both "valid" and "invalid" data. These transactions were based on specified procedures and regulations and were designed to check the effectiveness of internal controls in each installation's payroll processing. We used one transaction for each master record chosen.

The best method of obtaining suitable payroll master records for our test, we decided, would be to use copies of actual master records, supplemented with simulated records tailored for test conditions not found in the copied records.

Accordingly, we obtained a duplicate of each agency's payroll master file and had a section of it

APPENDIX I

printed in readable copy. From this printout, we selected a specific master record to go with each test transaction. When none of the copied records appearing on the printout fitted the specifics of a particular transaction, we made up a simulated master record by preparing source documents and processing them with the program used by each installation to add records for new employees to its master file. We then added the simulated records to the copied records to create our test master file.

We next prepared working papers on which were entered, for each test transaction, the control number assigned to the transaction, the type of input document to be used, and the nature and purpose of the test. We predetermined the correct end results for all test transactions and recorded these results in our working papers for comparison with actual results.

With some help from payroll office personnel, we next coded the test transactions onto source documents. The data was then keypunched into punched cards and key verified. We then processed the test data against actual agency payroll programs and

compared the te
results to see

We found b
several invalid
been rejected o
trols. Alterna
nonexistent or
they could be b
fraud, neglect,
mended that the
strengthened to
the Government

APPENDIX I

compared the test results with the predetermined results to see whether there were any differences.

We found both systems accepted and processed several invalid test transactions that should have been rejected or flagged by programmed computer controls. Alternative manual controls were either nonexistent or less than fully effective because they could be bypassed or compromised through fraud, neglect, or inadvertent error. We recommended that the systems' automated controls be strengthened to insure accurate payrolls and protect the Government from improper payments.

APPENDIX II

TYPICAL PAYROLL TRANSACTIONS TO BE
INCLUDED IN A TEST DECK

How a system with e

Rej
Print ce
error ci
Reject message st

Nature of test
transaction

Purpose or explanation
of test transaction

1. Leave a mandatory field blank on a new employee's master record.

To determine whether the system will accept a master record with essential data missing. If missing data will cause an incorrect payment, the master record should be rejected with appropriate warning; if missing data is for administrative purposes only, the condition should be flagged by an error message.

X

APPENDIX II

How a system with effective controls will handle the transaction

<u>Reject</u>	<u>Print error message</u>	<u>Reject in certain circumstances</u>	<u>Cut back to allowable maximum</u>	<u>Process without cutback</u>	<u>Automatically compute correct amount</u>	<u>Automatically adjust leave records</u>
	X	X				

APPENDIX II

<u>Nature of test transaction</u>	<u>Purpose or explanation of test transaction</u>	<u>How a system will</u>	
		<u>Reject</u>	<u>Print error message</u>
<p>2. Enter erroneous codes, such as charity, life insurance, union dues, marital status, etc. (Note--one erroneous code per master record.)</p>	<p>To determine whether the system will accept invalid data into employees' master records. The program should print error messages to identify invalid data and reject further processing of such transactions.</p>		X
<p>3. Enter an invalid annual leave category.</p>	<p>To determine whether the system will accept an invalid annual leave category. Federal regulations have established annual leave categories as 4, 6, or 8 depending on the amount of creditable service.</p>	X	X

APPENDIX II

How a system with effective controls will handle the transaction

<u>Reject</u>	<u>Print error message</u>	<u>Reject in certain circumstances</u>	<u>Cut back to allowable maximum</u>	<u>Process without cutback</u>	<u>Automatically compute correct amount</u>	<u>Automatically adjust leave records</u>
---------------	----------------------------	--	--------------------------------------	--------------------------------	---	---

	X	X				
--	---	---	--	--	--	--

X	X					
---	---	--	--	--	--	--

APPENDIX II

<u>Nature of test transaction</u>	<u>Purpose or explanation of test transaction</u>	<u>How a system with</u>	
		<u>Reject</u>	<u>Print error message</u> ^R
4. Change a field in an inactive master record.	To determine whether it is possible to change fields in inactive master records and whether adequate controls exist over such changes. Processing of inactive records should be separated from the normal processing of active records to eliminate the possibility of unearned salary payments or the manipulation of records for persons who are not in a pay status.	X	X

APPENDIX II

How a system with effective controls will handle the transaction

<u>Reject</u>	<u>message</u>	<u>Reject in certain circumstances</u>	<u>Cut back to allowable maximum</u>	<u>Process without cutback</u>	<u>Automatically compute correct amount</u>	<u>Automatically adjust leave records</u>
X	X					

APPENDIX II

A.

How a system with e

<u>Nature of test transaction</u>	<u>Purpose or explanation of test transaction</u>	<u>Reject</u>	<u>message</u>	<u>Rej ce ci st.</u>
5. Change an employee's annual leave category before it is due to be changed.	To determine whether the system will reject invalid updates. The annual leave category is based on the amount of creditable service an employee has, computed from the employee's service computation date. Employees with less than 3 years of service are in category 4; employees with 3 to 15 years of service are in category 6; employees with more than 15 years of service are in category 8. The program should reject any attempt to change a leave category before it is due to be changed.	X	X	

APPENDIX II

How a system with effective controls will handle the transaction

<u>Reject</u>	<u>Print error message</u>	<u>Reject in certain circumstances</u>	<u>Cut back to allowable maximum</u>	<u>Process without cutback</u>	<u>Automatically compute correct amount</u>	<u>Automatically adjust leave records</u>
	X					X

APPENDIX II

<u>Nature of test transaction</u>	<u>Purpose or explanation of test transaction</u>	<u>How a system with eff</u>	
		<u>Reject</u>	<u>Print error message</u> ^{or} <u>Rejec cert circ stan</u>
6. Promote a general schedule (GS) employee above grade 5 before 1 year in grade has passed.	To determine whether the system rejects an invalid transaction. Federal regulations state that GS employees above grade 5 must be in grade at least 1 year before they can be promoted.	X	X
7. Give a GS employee a within-grade salary increase before 1 year in grade has passed.	To determine how the system handles this transaction. Federal regulations state that a GS employee must be in grade at least 1 year before being eligible for a within-grade salary increase. The system should "flag" the transaction as being a quality step increase (which has the same effect as a within-grade increase but can occur without the employee's having been in grade for 1 year).		X

APPENDIX II

How a system with effective controls will handle the transaction

<u>Reject</u>	<u>Print error message</u>	<u>Reject in certain circumstances</u>	<u>Cut back to allowable maximum</u>	<u>Process without cutback</u>	<u>Automatically compute correct amount</u>	<u>Automatically adjust leave records</u>
X	X					

X

APPENDIX II

<u>Nature of test transaction</u>	<u>Purpose or explanation of test transaction</u>	<u>How a system with</u>										
		<table border="1"> <tr> <td></td> <td>Re</td> </tr> <tr> <td></td> <td>Print</td> </tr> <tr> <td></td> <td>error</td> </tr> <tr> <td><u>Reject</u></td> <td><u>message</u></td> </tr> <tr> <td></td> <td><u>st</u></td> </tr> </table>		Re		Print		error	<u>Reject</u>	<u>message</u>		<u>st</u>
	Re											
	Print											
	error											
<u>Reject</u>	<u>message</u>											
	<u>st</u>											
<p>8. Change an employee's grade or annual salary so that the grade/step and annual salary rate are incompatible.</p>	<p>To determine whether the system accepts incompatible data. The program should have salary and grade controls which will reject transactions of this type from further processing (except for payments under the Civil Service "retained rate" provision, which allows certain downgraded employees to retain their old salaries for a time).</p>	X										
<p>9. Change an employee's service computation date to indicate that leave category is due to change.</p>	<p>To determine whether the annual leave category is correctly changed, with a message printed to indicate the change. If the leave category is not automatically changed, a message should be printed.</p>	X										

APPENDIX II

How a system with effective controls will handle the transaction

<u>Reject</u>	<u>Print error message</u>	<u>Reject in certain circumstances</u>	<u>Cut back to allowable maximum</u>	<u>Process without cutback</u>	<u>Automatically compute correct amount</u>	<u>Automatically adjust leave records</u>
	X	X				

X

APPENDIX II

<u>Nature of test transaction</u>	<u>Purpose or explanation of test transaction</u>	<u>How a system with</u>	
		<u>Reject</u>	<u>Print error message</u>
10. Pay an inactive employee.	To determine whether the system will compute pay for an inactive employee (an employee who has been separated but whose record is maintained in the same master file used for active employees).	X	X
11. Pay a nonexistent employee	To determine whether the system will compute pay for an employee with no record in the master file.	X	X
12. Input two time and attendance cards for the same employee.	To determine whether the system will compute pay twice for the same employee.	X	X

APPENDIX II

How a system with effective controls will handle the transaction

<u>Reject</u>	<u>Print error message</u>	<u>Reject in certain circumstances</u>	<u>Cut back to allowable maximum</u>	<u>Process without cutback</u>	<u>Automatically compute correct amount</u>	<u>Automatically adjust leave records</u>
---------------	----------------------------	--	--------------------------------------	--------------------------------	---	---

X	X					
---	---	--	--	--	--	--

X	X					
---	---	--	--	--	--	--

X	X					
---	---	--	--	--	--	--

APPENDIX II

<u>Nature of test transaction</u>	<u>Purpose or explanation of test transaction</u>	<u>How a system with e</u>	
		<u>Reject</u>	<u>Print error message</u>
13. Pay a GS employee for 80 hours work on a second-shift entitlement for a wage board (WB) employee.	To determine whether the system rejects WB entitlements for GS employees.	X	X
14. Pay a GS employee for 80 hours work on a third-shift entitlement for a WB employee.	Same as above.	X	X
15. Pay a WB employee for 80 hours work on a night-differential entitlement for a GS employee.	To determine whether the system rejects GS entitlements for WB employees.	X	X

APPENDIX II

How a system with effective controls will handle the transaction

<u>Reject</u>	<u>Print error message</u>	<u>Reject in certain circumstances</u>	<u>Cut back to allowable maximum</u>	<u>Process without cutback</u>	<u>Automatically compute correct amount</u>	<u>Automatically adjust leave records</u>
	X					X
	X					X
	X					X

APPENDIX II

How a system with e

Rej
 Print ce.
 error ci:
Reject message sta

<u>Nature of test transaction</u>	<u>Purpose or explanation of test transaction</u>
16. Pay a WB employee for 20 hours of overtime.	To verify the accuracy of premium (overtime) pay computation. Overtime pay is 1-1/2 times regular pay.
17. Pay a GS employee for 20 hours of night-differential pay.	Same as above. Premium = 10 percent.
18. Pay a WB employee for 80 hours on second shift.	Same as above. Premium = 7-1/2 percent.
19. Pay a WB employee for 80 hours on third shift.	Same as above. Premium = 10 percent.

APPENDIX II

How a system with effective controls will handle the transaction

<u>Reject</u>	<u>message</u>	<u>Reject in</u> certain circum- stances	<u>Cut back</u> to allowable maximum	<u>Process</u> without cutback	<u>Automat-</u> <u>ically</u> compute correct amount	<u>Automat-</u> <u>ically</u> adjust leave records
---------------	----------------	---	---	--------------------------------------	--	--

X

X

X

X

APPENDIX II

How a system with effecti

Reject in
 Print certain
 error circum-
Reject message stances

<u>Nature of test transaction</u>	<u>Purpose or explanation of test transaction</u>
20. Pay a GS employee for 8 hours of holiday pay.	Same as above. Holiday pay is double regular pay.
21. Pay a WB employee for 8 hours of holiday pay.	Same as above.
22. Pay a GS employee for 8 hours of Sunday pay (for Sunday work that is not overtime work).	Same as above. Sunday premium is 25 percent of regular pay if Sunday is a regularly scheduled workday.
23. Pay a WB employee for 8 hours of Sunday pay.	Same as above.

APPENDIX II

How a system with effective controls will handle the transaction

<u>Reject</u>	<u>Print error message</u>	<u>Reject in certain circumstances</u>	<u>Cut back to allowable maximum</u>	<u>Process without cutback</u>	<u>Automatically compute correct amount</u>	<u>Automatically adjust leave records</u>
---------------	----------------------------	--	--------------------------------------	--------------------------------	---	---

X

X

X

X

APPENDIX II

<u>Nature of test transaction</u>	<u>Purpose or explanation of test transaction</u>
<p>24. Pay GS employees for 10 hours of environmental pay at the following premiums:</p> <ul style="list-style-type: none"> a. 4% b. 8% c. 25% d. 50% 	<p>Same as above.</p>
<p>25. Pay WB employees for 10 hours of environmental pay at the following premiums:</p> <ul style="list-style-type: none"> a. 4% b. 8% c. 25% d. 50% 	<p>Same as above.</p>

How a system with effective

<u>Reject</u>	<u>Print error message</u>	<u>Reject in certain circumstances</u>
---------------	----------------------------	--

APPENDIX II

How a system with effective controls will handle the transaction

<u>Reject</u>	<u>message</u>	<u>Reject in</u> <u>certain</u> <u>circum-</u> <u>stances</u>	<u>Cut back</u> <u>to</u> <u>allowable</u> <u>maximum</u>	<u>Process</u> <u>without</u> <u>cutback</u>	<u>Automat-</u> <u>ically</u> <u>compute</u> <u>correct</u> <u>amount</u>	<u>Automat-</u> <u>ically</u> <u>adjust</u> <u>leave</u> <u>records</u>
---------------	----------------	--	--	--	---	---

X

X

APPENDIX II

<u>Nature of test transaction</u>	<u>Purpose or explanation of test transaction</u>
26. Pay a GS-1, 2, 3, 4, 5, 6, or 7 employee for 10 hours of overtime.	To verify accuracy of premium pay computation. For GS employees whose basic pay rate does not exceed the salary of a GS-10/1, the overtime rate is 1-1/2 times the basic pay rate. (FPM 550-5)
27. Pay a GS-10, 11, 12, or 13 employee for 10 hours of overtime.	Same as above. For a GS employee whose basic pay rate is equal to or exceeds the rate of a GS-10/1, the overtime rate is 1-1/2 times the hourly rate for a GS-10/1. (FPM 550-5)

How a system with effect

	Reject i
Print	certain
error	circum-
<u>Reject</u>	<u>message</u> <u>stances</u>

APPENDIX II

How a system with effective controls will handle the transaction

<u>Reject</u>	<u>message</u>	<u>Reject in</u> certain circum- stances	<u>Cut back</u> to allowable maximum	<u>Process</u> without cutback	<u>Automat-</u> ically compute correct amount	<u>Automat-</u> ically adjust leave records
---------------	----------------	---	---	--------------------------------------	---	---

X

X

APPENDIX II

<u>Nature of test transaction</u>	<u>Purpose or explanation of test transaction</u>	<u>How a system with effecti</u>	
		<u>Reject</u>	<u>Print error message</u> <u>Reject in certain circumstances</u>
28. Pay a GS-14 employee enough overtime pay to exceed the maximum salary limitation.	To test maximum salary limitation. Additional pay, such as overtime, night-differential, holiday, and Sunday pay, may be paid to the extent that it does not cause the aggregate pay for a biweekly period to exceed the rate of a GS-15/10. (FPM, 550.105) The program should cut back pay to this maximum.	X	
29. Pay a GS-14 employee enough environmental pay to exceed the maximum salary limitation.	Same as above. Program should not cut back environmental pay because it is not subject to the maximum salary limitation.		

APPENDIX II

How a system with effective controls will handle the transaction

<u>Reject</u>	<u>Print error message</u>	<u>Reject in certain circumstances</u>	<u>Cut back to allowable maximum</u>	<u>Process without cutback</u>	<u>Automatically compute correct amount</u>	<u>Automatically adjust leave records</u>
	X		X			
				X		

APPENDIX II

How a system with effect

Reject :
 Print certain
 error circum:
Reject message stance.

<u>Nature of test transaction</u>	<u>Purpose or explanation of test transaction</u>
30. Pay a WB employee enough premium pay to exceed the maximum salary limitation.	Same as above. Program should not cut back pay because WB employees have no maximum salary limitation.
31. Pay a GS employee for 1 hour of holiday pay.	To determine whether the system will pay less than the 2-hour minimum of holiday pay. (FPM 990-1)
32. Pay a WB employee for 1 hour of holiday pay.	Same as above.
33. Pay a GS employee for 40 hours of Sunday pay.	To determine whether the system limits Sunday pay to 32 hours maximum allowed. (FPM 990-2)

X

X

X

APPENDIX II

How a system with effective controls will handle the transaction

<u>Reject</u>	<u>Print error message</u>	<u>Reject in certain circumstances</u>	<u>Cut back to allowable maximum</u>	<u>Process without cutback</u>	<u>Automatically compute correct amount</u>	<u>Automatically adjust leave records</u>
---------------	----------------------------	--	--------------------------------------	--------------------------------	---	---

X

X

X

X

X

X

X

APPENDIX II

<u>Nature of test transaction</u>	<u>Purpose or explanation of test transaction</u>
<p>34. Pay a WB employee for 80 hours on second shift and 10 hours for overtime into the third shift.</p>	<p>To verify the accuracy of premium pay. Federal regulations state that overtime pay for an employee regularly working the second or third shift will be computed at 1-1/2 times the second or third shift rate, respectively. (FPM 532-1)</p>
<p>35. Pay a WB employee for 80 hours on third shift and 10 hours for overtime into the first shift.</p>	<p>Same as above.</p>

How a system with effect

<u>Reject</u>	<u>message</u>	<u>Reject i Print certain error circum- stances</u>
---------------	----------------	---

APPENDIX II

How a system with effective controls will handle the transaction

<u>Reject</u>	<u>message</u>	<u>Reject in</u> Print error stances	<u>Cut back</u> certain circum- allowable maximum	<u>Process</u> without cutback	<u>Automat-</u> ically compute correct amount	<u>Automat-</u> ically adjust leave records
---------------	----------------	---	---	--------------------------------------	---	---

X

X

u-
.me
-
or
uted
d or

APPENDIX II

<u>Nature of test transaction</u>	<u>Purpose or explanation of test transaction</u>	<u>How a system with effective</u>			
		<table border="1"> <tr> <td></td> <td data-bbox="1325 664 1404 750">Print error message</td> <td data-bbox="1429 633 1572 750">Reject in certain circumstances</td> </tr> </table>		Print error message	Reject in certain circumstances
	Print error message	Reject in certain circumstances			
<p>36. Charge a full-time employee for 80 hours of leave without pay (LWOP).</p>	<p>To determine whether sick and annual leave will accrue when a full-time employee charges 80 hours LWOP. The sick leave credit should be reduced by 4 hours and the annual leave credit should be reduced by 4, 6, or 8 hours, depending on the annual leave category.</p>				
<p>37. Charge a full-time employee for more annual leave than the employee has.</p>	<p>To determine whether excess annual leave is charged to LWOP. (The system should automatically reduce employee's pay for LWOP.)</p>	X			

APPENDIX II

How a system with effective controls will handle the transaction

<u>Reject</u>	<u>Print error message</u>	<u>Reject in certain circumstances</u>	<u>Cut back to allowable maximum</u>	<u>Process without cutback</u>	<u>Automatically compute correct amount</u>	<u>Automatically adjust leave records</u>
						X
	X				X	X

ue

he

de

le

d

lly

r

APPENDIX II

<u>Nature of test transaction</u>	<u>Purpose or explanation of test transaction</u>	<u>How a system with effect</u>	
		<u>Reject</u>	<u>Print error message</u> <u>Reject certain circur stance</u>
38. Charge a full-time employee for more sick leave than the employee has.	To determine whether excess sick leave is charged to annual leave or LWOP. (The system should automatically adjust leave records and reduce pay for LWOP, if required.)	X	
39. Charge a full-time employee for 99 hours of annual leave (19 hours more than a regular biweekly period).	To determine whether the system will cut back to the maximum of 80 hours for regular pay in a pay period.		X
40. Charge a full-time employee for 99 hours of sick leave.	Same as above.		X

APPENDIX II

How a system with effective controls will handle the transaction

<u>Reject</u>	<u>Print error message</u>	<u>Reject in certain circumstances</u>	<u>Cut back to allowable maximum</u>	<u>Process without cutback</u>	<u>Automatically compute correct amount</u>	<u>Automatically adjust leave records</u>
---------------	----------------------------	--	--------------------------------------	--------------------------------	---	---

X

X

X

X

X

X

X

APPENDIX II

<u>Nature of test transaction</u>	<u>Purpose or explanation of test transaction</u>	<u>How a system with effective control</u>			
		<u>Reject</u>	<u>Print error message</u>	<u>Reject in certain circumstances</u>	<u>Cu all ma</u>
41. Charge a full-time employee for 80 hours of regular pay and 80 hours of annual leave in the same pay period.	Same as above. Total hours of work and leave cannot exceed 80 in a pay period.	X	X		
42. Charge a full-time employee for enough hours of military leave to exceed 120 hours total.	To determine whether the system flags military leave in excess of 120 hours. Federal regulations state that an employee cannot charge more than 120 hours to military leave in a pay year. Because there are certain exceptions (such as duty in the District of Columbia National Guard) which permit military leave to exceed 120 hours, the				

APPENDIX II

How a system with effective controls will handle the transaction

<u>Reject</u>	<u>Print error message</u>	<u>Reject in certain circumstances</u>	<u>Cut back to allowable maximum</u>	<u>Process without cutback</u>	<u>Automatically compute correct amount</u>	<u>Automatically adjust leave records</u>
	X	X				

APPENDIX II

Nature of test transaction

Purpose or explanation of test transaction

How a system with effect:

Reject in:
 Print certain
 error circum-
Reject message stances

42. cont'd.

system should alert payroll clerks to the excess and should not reject or cut back the transaction.

43. Make a lump-sum annual leave payment to a separated employee in excess of annual leave balance.

To determine whether the system appropriately excludes excess annual leave in a lump-sum leave payment.

X

APPENDIX II

<u>Nature of test transaction</u>	<u>Purpose or explanation of test transaction</u>
44. Pay a GS part-time employee for 32 hours of regular pay.	To determine whether the system correctly accrues annual and sick leave for part-time employees. For each 20 hours worked, a part-time employee receives 1 hour of sick leave. If in leave category 4, an employee needs 20 hours of work to earn 1 hour of annual leave; if in leave category 6, the employee needs 15 hours worked to earn 1 hour of annual leave; and if in leave category 8, the employee needs 10 hours worked to earn 1 hour of annual leave.

How a system with effecti

Reject message Print error Reject in certain circum-
stances

APPENDIX II

How a system with effective controls will handle the transaction

	Reject in	Cut back		Automat-	Automat-
	Print	certain	to	ically	ically
	error	circum-	allowable	Process	compute
	message	stances	maximum	without	correct
<u>Reject</u>	<u>message</u>	<u>stances</u>	<u>maximum</u>	<u>cutback</u>	<u>amount</u>
					<u>records</u>

X

PPENDIX II

<u>Nature of test transaction</u>	<u>Purpose or explanation of test transaction</u>	<u>How a system with effective</u>		
		<u>Reject</u>	<u>Print error message</u>	<u>Reject in certain circumstances</u>
5. Make a lump-sum annual leave payment to an active employee.	To determine whether the system will make a lump-sum annual leave payment to an active employee. These payments should be made only to separated employees.	X	X	

APPENDIX II

How a system with effective controls will handle the transaction

<u>Reject</u>	<u>Print error message</u>	<u>Reject in certain circumstances</u>	<u>Cut back to allowable maximum</u>	<u>Process without cutback</u>	<u>Automatically compute correct amount</u>	<u>Automatically adjust leave records</u>
	X					

X

X