Software Testing

Project Report

1) Transition Pair Testing

State S0
Incoming: T1, T19, T5, T3, T14, T13
Outgoing: T2, T6, T20

Transition Pair	Covering Test Case
(T1, T2)	Test #3
(T1, T6)	Test #1
(T1, T20)	Test #4
(T3, T2)	**Test #3**
(T3, T6)	Test #5
(T3, T20)	Test #6
(T5, T2)	**Test #6**
(T5, T6)	Test #3
(T5, T20)	Test #1
(T13, T2)	**Test #1**
(T13, T6)	Test #7
(T13, T20)	Test #3
(T14, T2)	**Test #2**
(T14, T6)	Test #2
(T14, T20)	Test #5
(T19, T2)	**Test #6**
(T19, T6)	Test #5
(T19, T20)	Test #2

State S1
Incoming: T17, T18
Outgoing: T16, T17

Transition Pair	Covering Test Case
(T17, T16)	Test #2
(T17, T17)	Test #2

(T18, T16)	Test #7
(T18, T17)	Test #2

State S2
Incoming: T2
Outgoing: T3, T4

Transition Pair	**Covering Test Case**
(T2, T3)	Test #3
(T2, T4)	Test #1

State S3
Incoming: T4, T6
Outgoing: T5, T7, T8

Transition Pair	**Covering Test Case**
(T4, T5)	Test #1
(T4, T7)	Test #2
(T4, T8)	Test #6

(T6, T5)	**Test #5**
(T6, T7)	Test #1
(T6, T8)	Test #2

State S4
Incoming: T7, T8
Outgoing: T9, T18, T19

Transition Pair	**Covering Test Case**
(T7, T9)	Test #1
(T7, T18)	Test #2
(T7, T19)	Test #6

(T8, T9)	**Test #3**
(T8, T18)	Test #7
(T8, T19)	Test #2

State S5
Incoming: T9, T10
Outgoing: T10, T11, 12

Transition Pair	**Covering Test Case**

(T9, T10)	Test #1
(T9, T11)	Test #3
(T9, T12)	Test #2

(T10, T10)	**Test #1**
(T10, T11)	Test #7
(T10, T12)	Test #1

State S6
Incoming: T11, T12, T16
Outgoing: T13, T14

Transition Pair	**Covering Test Case**
(T11, T13)	Test #3
(T11, T14)	Test #7

(T12, T13)	**Test #1**
(T12, T14)	Test #2

(T16, T13)	**Test #7**
(T16, T14)	Test #2

Test Cases:

Test #1
Activate(6.5, 8.5), Cash(36.0), Regular(), Start(), Pump(), Pump(), Pump(), Pump(), Pump(), Receipt(), Credit(), Approved(), Cancel(), TurnOff()

Transition: T1 -> T6 -> T7 -> T9 -> T10 -> T10 -> T10 -> T10 -> T12 -> T13 -> T2 -> T4 -> T5 -> T20 ->

Test #2
Activate(3, 4), Cash(1), Regular(), Start(), Pump(), NoReceipt(), Credit(), Approved(), Regular(), Start(), Pump(), Pump(), Stop(), NoReceipt(), Cash(10.0), Diesel(), Cancel(), TurnOff()

Transition: T1 -> T6 -> T7 -> T9 -> T12 -> T14 -> T2 -> T4 -> T7 -> T18 -> T17 -> T17 -> T16 -> T14 -> T6 -> T8 -> T19 -> T20

Test #3
Activate(6, 4), Credit(), Reject(), Credit(), Approved(), Cancel(), Cash(10), Diesel(), Start(), Stop(), Receipt(), TurnOff()

Transition: T1 -> T2 -> T3 -> T2 -> T4 -> T5 -> T6 -> T8 -> T9 -> T11 -> T13 -> T20

Test #4
Activate(4, 5), TurnOff()

Transition: T1 -> T20

Test #5
Activate(3.0, 4.0), Credit(), Reject(), Cash(5.0), Cancel(), Cash(10), Diesel(), Cancel(), Cash(4), Regular(), Start(), Pump(), Stop(), NoReceipt(), TurnOff()

Transition: T1 -> T2 -> T3 -> T6 -> T5 -> T6 -> T8 -> T19 -> T6 -> T7 -> T9 -> T10 -> T12 -> T14 -> T20

Test #6
Activate(3.5, 4.0), Credit(), Approved(), Cancel(), Credit(), Approved(), Diesel(), Cancel(), Cash(1), Regular(), Cancel(),
Credit(), Reject(), TurnOff()

Transition: T1 -> T2 -> T4 -> T5 -> T2 -> T4 -> T8 -> T19 -> T6 -> T7 -> T19 -> T2 -> T3 -> T20 ->

Test #7
Activate(3.0, 4.0), Credit(), Approved(), Diesel(), Start(), Stop(), Receipt(), Cash(40), Diesel(), Start(),Pump(), Stop(), NoReceipt()

Transition: T1 -> T2 -> T4 -> T8 -> T18 -> T16 -> T13 -> (Print: 0 and 0) T6 -> T8 -> T9 -> T10 -> T11 -> T14

2) Default/Ghost Transitions Testing

State “Start”

Default Transition	**Covering Test Case**
Activate(a, d)[a <= 0 || d <= 0]	Test #8
Credit()	Test #8
Reject()	Test #8
Cancel()	Test #8
Approved()	Test #8
Cash(c)	Test #8
Regular()	Test #8
Diesel()	Test #8
Start()	Test #8
Pump()	Test #8
Stop()	Test #8
NoReceipt()	Test #8
Receipt()	Test #8
TurnOff()	Test #8

Test #8

Activate(-3.0, 4.0), Credit(), Reject(), Cancel(), Approved(), Cash(2), Regular(), Diesel(), Start(), Pump(), Stop(), NoReceipt(), Receipt(), TurnOff()

State "S0"

Default Transition	Covering Test Case
Activate(a, d)	Test #9
Reject()	Test #9
Cancel()	Test #9
Approved()	Test #9
Cash(c)[c <= 0]	Test #9
Regular()	Test #9
Diesel()	Test #9
Start()	Test #9
Pump()	Test #9
Stop()	Test #9
NoReceipt()	Test #9
Receipt()	Test #9

Test #9

Activate(3.0, 4.0), **Activate(-3.0, 4.0), Reject(), Cancel(), Approved(), Cash(-2), Regular(), Diesel(), Start(), Pump(), Stop(), NoReceipt(), Receipt()**

State "S1"

Default Transition	Covering Test Case
Activate(a, d)	Test #10
Credit()	Test #10
Reject()	Test #10
Cancel()	Test #10
Approved()	Test #10
Cash(c)	Test #10
Regular()	Test #10
Diesel()	Test #10
Start()	Test #10
NoReceipt()	Test #10
Receipt()	Test #10
TurnOff()	Test #10

Test #10

Activate(3.0, 4.0), Credit(), Approved(), Regular(), Start(), **Activate(-3.0, 4.0), Reject(), Cancel(), Approved(), Cash(-2), Regular(), Diesel(), Start(), NoReceipt(), Receipt()**

State "S2"

Default Transition	Covering Test Case
Activate(a, d)	Test #11
Credit()	Test #11
Cancel()	Test #11
Cash(c)	Test #11
Regular()	Test #11
Diesel()	Test #11
Start()	Test #11
Pump()	Test #11
Stop()	Test #11
NoReceipt()	Test #11
Receipt()	Test #11
TurnOff()	Test #11

Test #11 - changed to Activate(3.0, 4.0)
Activate(3.0, 4.0), Credit(), **Activate(3.0, 4.0), Credit(), Cancel(), Cash(2), Regular(), Diesel(), Start(), Pump(), Stop(), NoReceipt(), Receipt(), TurnOff()**

State "S3"

Default Transition	Covering Test Case
Activate(a, d)	Test #12
Credit()	Test #12
Reject()	Test #12
Approved()	Test #12
Cash(c)	Test #12
Start()	Test #12
Pump()	Test #12
Stop()	Test #12
NoReceipt()	Test #12
Receipt()	Test #12
TurnOff()	Test #12

Test #12
Activate(3.0, 4.0), Credit(), Approved(), **Activate(3.0, 4.0), Credit(), Reject(), Approved(), Cash(2), Start(), Pump(), Stop(), NoReceipt(), Receipt(), TurnOff()**

State "4"

Default Transition	Covering Test Case
Activate(a, d)	Test #13
Credit()	Test #13
Reject()	Test #13
Approved()	Test #13
Cash(c)	Test #13
Regular()	Test #13
Diesel()	Test #13
Start()[**!**(k == 0 || k == 1)]	Test #13 b
Pump()	Test #13
Stop()	Test #13
NoReceipt()	Test #13
Receipt()	Test #13
TurnOff()	Test #13

Test #13

Activate(3.0, 4.0), Credit(), Approved(), Regular(), **Activate(3.0, 4.0), Credit(), Reject(), Approved(), Cash(2), Regular(), Diesel(), Pump(), Stop(), NoReceipt(), Receipt(), TurnOff()**

Test #13 b

Activate(3.0, 4.0), Credit(), Approved(), Regular(), **Start() with internal k=2,**

State "S5"

Default Transition	Covering Test Case
Activate(a, d)	Test #14
Credit()	Test #14
Reject()	Test #14
Cancel()	Test #14
Approved()	Test #14
Cash(c)	Test #14
Regular()	Test #14
Diesel()	Test #14
Start()	Test #14
NoReceipt()	Test #14
Receipt()	Test #14
TurnOff()	Test #14

Test #14

Activate(3.0, 4.0), Cash(5),Regular(), Start()

Activate(3.0, 4.0), Credit(), Reject(), Cancel(), Approved(), Cash(2), Regular(), Diesel(), Start(), NoReceipt(), Receipt(), TurnOff()

State “S6”

Default Transition	Covering Test Case
Activate(a, d)	Test #15
Credit()	Test #15
Reject()	Test #15
Cancel()	Test #15
Approved()	Test #15
Cash(c)	Test #15
Regular()	Test #15
Diesel()	Test #15
Start()	Test #15
Pump()	Test #15
Stop()	Test #15
TurnOff()	Test #15

Test #15

Activate(3.0, 4.0), Cash(5),Regular(), Start(), Stop()

Activate(3.0, 4.0), Credit(), Reject(), Cancel(), Approved(), Cash(2), Regular(), Diesel(), Start(), Pump(), Stop(), TurnOff()

State “End”

Default Transition	Covering Test Case
Activate(a, d)	Test #16
Credit()	Test #16
Reject()	Test #16
Cancel()	Test #16
Approved()	Test #16
Cash(c)	Test #16
Regular()	Test #16
Diesel()	Test #16
Start()	Test #16
Pump()	Test #16
Stop()	Test #16
NoReceipt()	Test #16
Receipt()	Test #16

Test #16

Activate(3.0, 4.0), Cash(5),Regular(), Start(), Stop(), NoReceipt(), TurnOff(),

Activate(3.0, 4.0), Credit(), Reject(), Cancel(), Approved(), Cash(2), Regular(), Diesel(), Start(), Pump(), Stop(), NoReceipt(), Receipt(), TurnOff()

3) Multiple-conditional Testing

```
Activate(float a, float d)
        if ((k == -1) && (a > 0) && (d > 0))
```

k == -1	a > 0	d > 0	Test Coverage
T	T	T	Test #17
T	T	F	Test #19
T	F	T	Test #18
T	F	F	Test #20
F	T	T	Test #21
F	T	F	Test #23
F	F	T	Test #22
F	F	F	Test #24

Test #17
Activate(3.0, 4.0), Activate(3.0, 4.0), Cash(20), Cancel(), TurnOff()

Test #18
Activate(-3.0, 4.0), Activate(3.0, 4.0), Cash(20), Cancel(), TurnOff()

Test #19
Activate(3.0, -4.0), Activate(3.0, 4.0), Cash(20), Cancel(), TurnOff()

Test #20
Activate(-3.0, -4.0), Activate(3.0, 4.0), Cash(20), Cancel(), TurnOff()

Test #21
Activate(3.0, 4.0), **Activate(3.0, 4.0),** Cash(20), Cancel(), TurnOff()

Test #22
Activate(3.0, 4.0), **Activate(-3.0, 4.0),** Cash(20), Cancel(), TurnOff()

Test #23
Activate(3.0, 4.0), **Activate(3.0, -4.0),** Cash(20), Cancel(), TurnOff()

Test #24
Activate(3.0, 4.0), **Activate(-3.0, -4.0),** Cash(20), Cancel(), TurnOff()

```
public final int Credit() {
    if (k == 0) {
```

k == 0	Test Coverage
T	Test #25

F	Test #25

Test #25: Activate(3.0, 4.0), Activate(3.0, 4.0), Credit(), Credit(), Reject()

```
public final int Reject() {
    if (k == 2) {
```

k == 2	Test Coverage
T	Test #26
F	Test #26

Test #26: Activate(3.0, 4.0), Activate(3.0, 4.0), Credit(), Credit(), Reject(), Reject()

```
public final int Cancel() {
    if ((k == 3) || (k == 4)) {
```

k == 3	k == 4	Test Coverage
T	T	Not Executable
T	F	Test #27
F	T	Test #29
F	F	Test #28

```
        if (w == 0) {
```

w == 0	Test Coverage
T	Test #27
F	Test #30

Test #27: Activate(3.0, 4.0), Activate(3.0, 4.0), Credit(), Credit(), Approved(), Cancel();

Test #28: Activate(3.0, 4.0), Activate(3.0, 4.0), Credit(), Credit(), Approved(), Cancel(), Cancel()

Test #29: Activate(5.0, 6.0), Credit(), Approved(), Regular(), Start(), Pump(), Pump(), Stop(), Receipt(), Cash(20), Regular(), Cancel(), Credit()

Test #30: Activate(5.0, 6.0), Credit(), Approved(), Cancel()

Condition (T T) is not executable because "k" can't be 2 values at the same time hence, these two conditions can't be True at the same time

```
public final int Approved() {
    if (k == 2) {
```

k == 2	Test Coverage

T	Test #27
F	Test #31

Test #31: Activate(3.5, 4.5), Approved(), Cash(-20), Regular(), Receipt() , TurnOff()

```
public final int Cash(float c) {
    if (k == 0) {
```

k == 0	Test Coverage
T	Test #31
F	Test #32

Test #32: Activate(3.3 4.4), Credit(), Approved(), Regular(), Start(), Cash(-20), Stop(), NoReceipt(), TurnOff()

```
public final int Regular() {
    if (k == 3) {
```

k == 3	Test Coverage
T	Test #32
F	Test #33

Test #33: Activate(2.6, 3.1), Credit(), Regular(), Diesel(), Start(), Approved(), Regular(), Cancel(), TurnOff()

```
        if (w==1)
```

w == 1	Test Coverage
T	Test #33
F	Test #31

```
public final int Diesel() {
    if (k == 3) {
```

k == 3	Test Coverage
T	Test #34
F	Test #33

Test #34: Activate(5.0, 6.0), Credit(), Approved(), Regular(), Start(), Pump(), Pump(), Stop(), Receipt(), Cash(20), Diesel()

```
public final int Start() {
    if (k == 4) {
```

k == 4	Test Coverage
T	Test #34
F	Test #33

```
public final int Pump() {
    if (k == 5) {
```

k == 5	Test Coverage
T	Test #35
F	Test #36

Test #35: Activate(5.0, 6.0), Credit(), Approved(), Regular(), Start(), Pump(), Stop();

Test #36: Activate(3, 3.5), Pump(), Stop(), NoReceipt(), TurnOff()

```
        if ((w == 1) || ((cash > price * (L + 1)) && (w == 0))) {
```

w == 1	cash > price * (L + 1)	w == 0	Test coverage
T	T	T	Not Executable, w can't be 1 and 0 at same time
T	T	F	Test #37
T	F	T	Not Executable, w can't be 0 and 1 same time
T	F	F	Test #35
F	T	T	Test #38
F	T	F	Not Executable, both (w==0) and (w==1) can't be both False.
F	F	T	Test #39
F	F	F	Not Executable, both (w==0) and (w==1) can't be both False.

Test #37: Activate(3, 4), Cash(20), Cancel(), Credit(), Approved(), Regular(), Start(), Pump(), Stop()

Test #38: Activate(3.0f, 4.0f), Credit(), Reject(), gp.Cash(5); gp.Cancel(); gp.Cash(10); gp.Diesel(); gp.Cancel(); gp.Cash(4); gp.Regular();
gp.Start(); gp.Pump(); gp.Stop(); gp.NoReceipt(); gp.TurnOff();

Test #39: Activate(3.3, 4.4), Cash(1.1f), Regular(), Start(), Pump(), NoReceipt(), Credit();

```
        else if ((w == 0) && (cash < price * (L + 1))) {
```

w == 0	cash < price * (L + 1)	Test Coverage
T	T	Test #39
T	F	Test #38
F	T	Not Executable. If (w==0) evaluates to False, then (w==0) must be True. With (w==0) being True, the previous if statement if ((w == 1) || ((cash > price * (L + 1)) && (w == 0))) must be evaluated to True, which makes this else if statement unreachable. (re-Confirm)
F	F	Not Executable. If (w==0) evaluates to False, then (w==0) must be True. With (w==0) being True, the previous if statement if ((w == 1) || ((cash > price * (L + 1)) && (w == 0))) must be evaluated to True, which makes this else if statement unreachable. (re-Confirm)

```
        if ((w == 0) && (total < cash)) {
```

w == 0	total < cash	Test Coverage
T	T	Test #39
T	F	Test #41
F	T	if ((w == 0) && (total < cash)) is nested inside the previous else if statement. Therefore, when the execution reaches this if statement, (w==0) must be True.
F	F	if ((w == 0) && (total < cash)) is nested inside the previous else if statement. Therefore, when the execution reaches this if statement, (w==0) must be True.

```
public final int Stop() {
    if (k == 5) {
```

k == 5	Test Coverage
T	Test #38
F	Test #40

Test #40: Activate(3.0f, 4.0f), Cash(2.0f), Diesel(), Start(); Pump(); Pump(); Pump(); Pump(), Stop();

Test #41: Activate(3.0f, 4.0f); Credit(); Reject(); Cash(5); Cancel(); Cash(10); Diesel(); Cancel(); Cash(-3); Regular(); Start(); Pump(); Stop(); NoReceipt(); TurnOff();

```
        if ((w == 0) && (total < cash)) {
```

w == 0	total < cash	Test Coverage
T	T	Test #1
T	F	Not executable, cut total doesn't get updated [confirm]
F	T	Test #37
F	F	Test #35

```
public final int NoReceipt() {
    if (k == 6) {
```

k == 6	Test Coverage
T	Test #38
F	Test #36

```
public final int Receipt() {
    if (k == 6) {
```

k == 6	Test Coverage
T	Test #34
F	Test #31

```
public final int TurnOff() {
    if (k == 0) {
```

k == 0	Test Coverage
T	Test #41
F	Test #42

Test #42: Activate(3.0f, 4.0f); Credit(); Approved(); Diesel(); Start(); Pump(); Pump(); Receipt(); TurnOff();

4) Test Suite

Test#1: Activate 6.5 8.5 Cash 36.0 Regular Start Pump Pump Pump Pump Pump Receipt Credit Approved Cancel TurnOff
Test#2: Activate 3 4 Cash 1 Regular Start Pump NoReceipt Credit Approved Regular Start Pump Pump Stop NoReceipt Cash 10.0 Diesel Cancel TurnOff
Test#3: Activate 6 4 Credit Reject Credit Approved Cancel Cash 10 Diesel Start Stop Receipt TurnOff
Test#4: Activate 4 5 TurnOff
Test#5: Activate 3.0 4.0 Credit Reject Cash 5.0 Cancel Cash 10 Diesel Cancel Cash 4 Regular Start Pump Stop NoReceipt TurnOff
Test#6: Activate 3.5 4.0 Credit Approved Cancel Credit Approved Diesel Cancel Cash 1 Regular Cancel Credit Reject TurnOff
Test#7: Activate 3.0 4.0 Credit Approved Diesel Start Stop Receipt Cash 40 Diesel Start Pump Stop NoReceipt
Test#8: Activate -3.0 4.0 Credit Reject Cancel Approved Cash 2 Regular Diesel Start Pump Stop NoReceipt Receipt TurnOff
Test#9: Activate 3.0 4.0 Activate -3.0 4.0 Reject Cancel Approved Cash -2 Regular Diesel Start Pump Stop NoReceipt Receipt
Test#10: Activate 3.0 4.0 Credit Approved Regular Start Activate -3.0 4.0 Reject Cancel Approved Cash -2 Regular Diesel Start NoReceipt Receipt
Test#11: Activate 3.0 4.0 Credit Activate 3.0 4.0 Credit Cancel Cash 2 Regular Diesel Start Pump Stop NoReceipt Receipt TurnOff
Test#12: Activate 3.0 4.0 Credit Approved Activate 3.0 4.0 Credit Reject Approved Cash 2 Start Pump Stop NoReceipt Receipt TurnOff
Test#13: Activate 3.0 4.0 Credit Approved Regular Activate 3.0 4.0 Credit Reject Approved Cash 2 Regular Diesel Pump Stop NoReceipt Receipt TurnOff
Test#14: Activate 3.0 4.0 Cash 5 Regular Start Activate 3.0 4.0 Credit Reject Cancel Approved Cash 2 Regular Diesel Start NoReceipt Receipt TurnOff
Test#15: Activate 3.0 4.0 Cash 5 Regular Start Stop Activate 3.0 4.0 Credit Reject Cancel Approved Cash 2 Regular Diesel Start Pump Stop TurnOff
Test#16: Activate 3.0 4.0 Cash 5 Regular Start Stop NoReceipt TurnOff Activate 3.0 4.0 Credit Reject Cancel Approved Cash 2 Regular Diesel Start Pump Stop NoReceipt Receipt TurnOff
Test#17: Activate 3.0 4.0 Activate 3.0 4.0 Cash 20 Cancel TurnOff
Test#18: Activate -3.0 4.0 Activate 3.0 4.0 Cash 20 Cancel TurnOff
Test#19: Activate 3.0 -4.0 Activate 3.0 4.0 Cash 20 Cancel TurnOff
Test#20: Activate -3.0 -4.0 Activate 3.0 4.0 Cash 20 Cancel TurnOff
Test#21: Activate 3.0 4.0 Activate 3.0 4.0 Cash 20 Cancel TurnOff
Test#22: Activate 3.0 4.0 Activate -3.0 4.0 Cash 20 Cancel TurnOff
Test#23: Activate 3.0 4.0 Activate 3.0 -4.0 Cash 20 Cancel TurnOff
Test#24: Activate 3.0 4.0 Activate -3.0 -4.0 Cash 20 Cancel TurnOff
Test#25: Activate 3.0 4.0 Activate 3.0 4.0 Credit Credit Reject
Test#26: Activate 3.0 4.0 Activate 3.0 4.0 Credit Credit Reject Reject
Test#27: Activate 3.0 4.0 Activate 3.0 4.0 Credit Credit Approved Cancel
Test#28: Activate 3.0 4.0 Activate 3.0 4.0 Credit Credit Approved Cancel Cancel
Test#29: Activate 5.0 6.0 Credit Approved Regular Start Pump Pump Stop Receipt Cash 20 Regular Cancel Credit

Test#30: Activate 5.0 6.0 Credit Approved Cancel
Test#31: Activate 3.5 4.5 Approved Cash -20 Regular Receipt TurnOff
Test#32: Activate 3.3 4.4 Credit Approved Regular Start Cash -20 Stop NoReceipt TurnOff
Test#33: Activate 2.6 3.1 Credit Regular Diesel Start Approved Regular Cancel TurnOff
Test#34: Activate 5.0 6.0 Credit Approved Regular Start Pump Pump Stop Receipt Cash 20 Diesel
Test#35: Activate 5.0 6.0 Credit Approved Regular Start Pump Stop
Test#36: Activate 3 3.5 Pump Stop NoReceipt TurnOff
Test#37: Activate 3 4 Cash 20 Cancel Credit Approved Regular Start Pump Stop
Test#38: Activate 3.0 4.0 Credit Reject Cash 5 Cancel Cash 10 Diesel Cancel Cash 4 Regular Start Pump Stop NoReceipt TurnOff
Test#39: Activate 3.3 4.4 Cash 1.1 Regular Start Pump NoReceipt Credit
Test#40: Activate 3.0 4.0 Cash 2.0 Diesel Start Pump Pump Pump Pump Stop
Test#41: Activate 3.0 4.0 Credit Reject Cash 5 Cancel Cash 10 Diesel Cancel Cash -3 Regular Start Pump Stop NoReceipt TurnOff
Test#42: Activate 3.0 4.0 Credit Approved Diesel Start Pump Pump Receipt TurnOff
$$

5) Test Results