Indian Institute of Information Technology, Guwahati

DD Pac-Man

SOFTWARE ENGINEERING PROJECT

Made by:-

1→Deepak Sidar (1801050)

2→Diwakar Bharti (1801059)

Note: This project is made by a student of BTech Computer Science and may contain some errors. You can Improve this software at this website CLICK HERE

Logo of DD Pac-Man

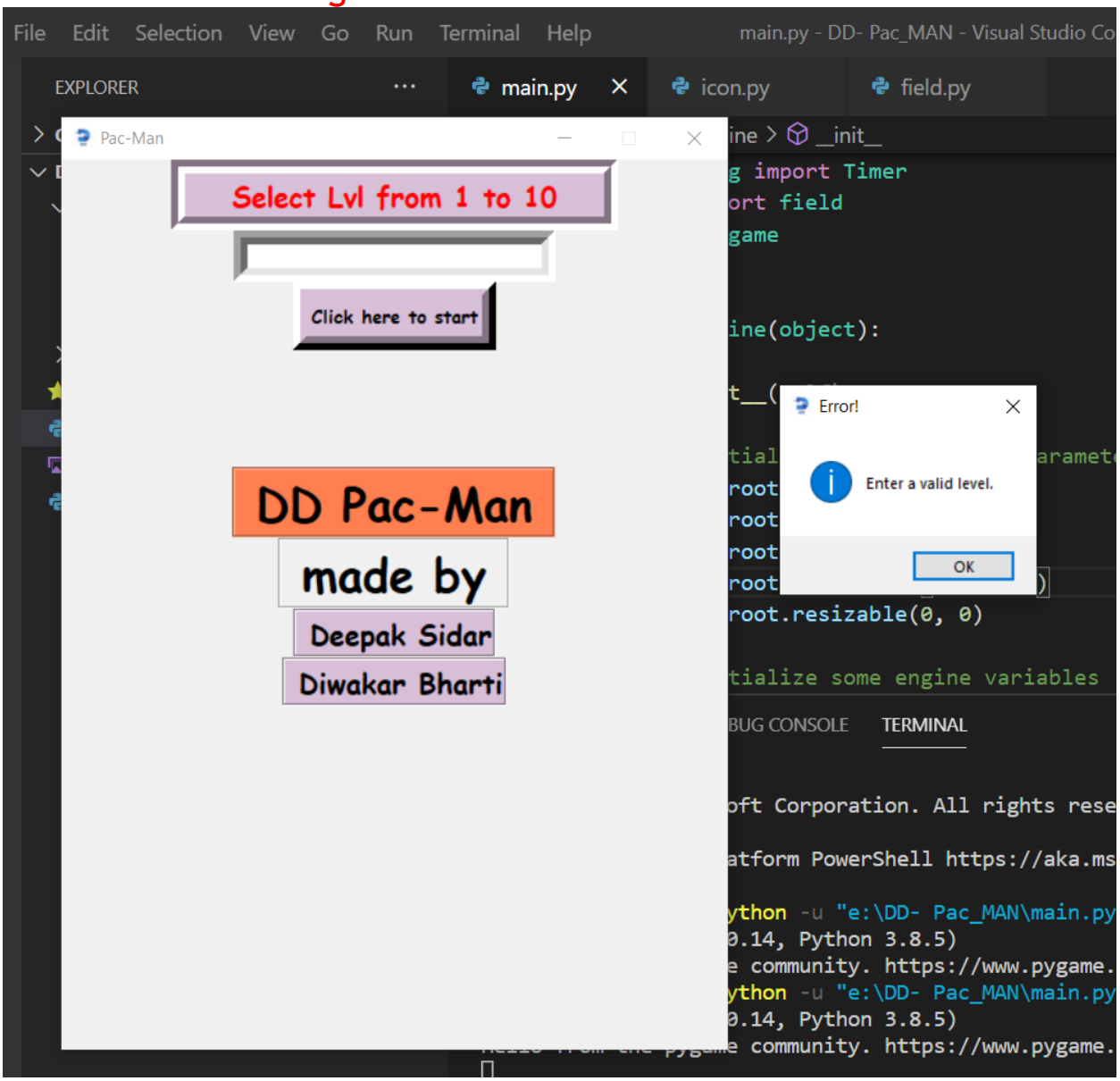
This logo is used as a root logo of game, as well as wall of game.

Functions of DD Pac-man are given below:-

1 User can select level from 1- 10

Here, user have to select the level which he/she want to play from (1-10). Some players are expert, So they can directly move to hard level.

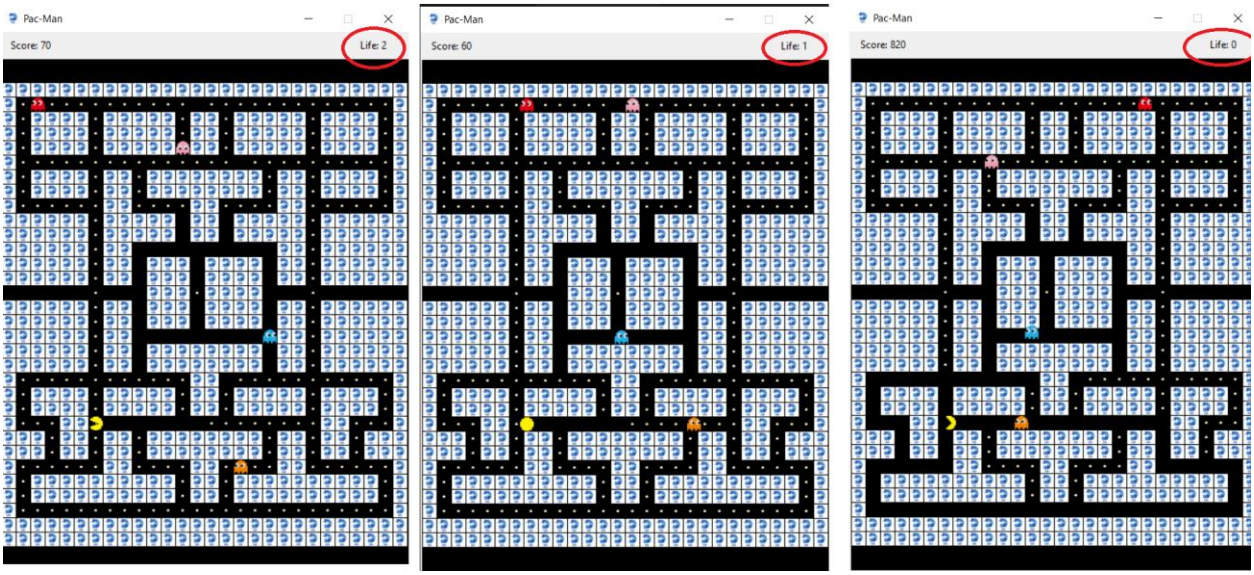
Black Box testing->



2 User are provided with three life to complete a Level.

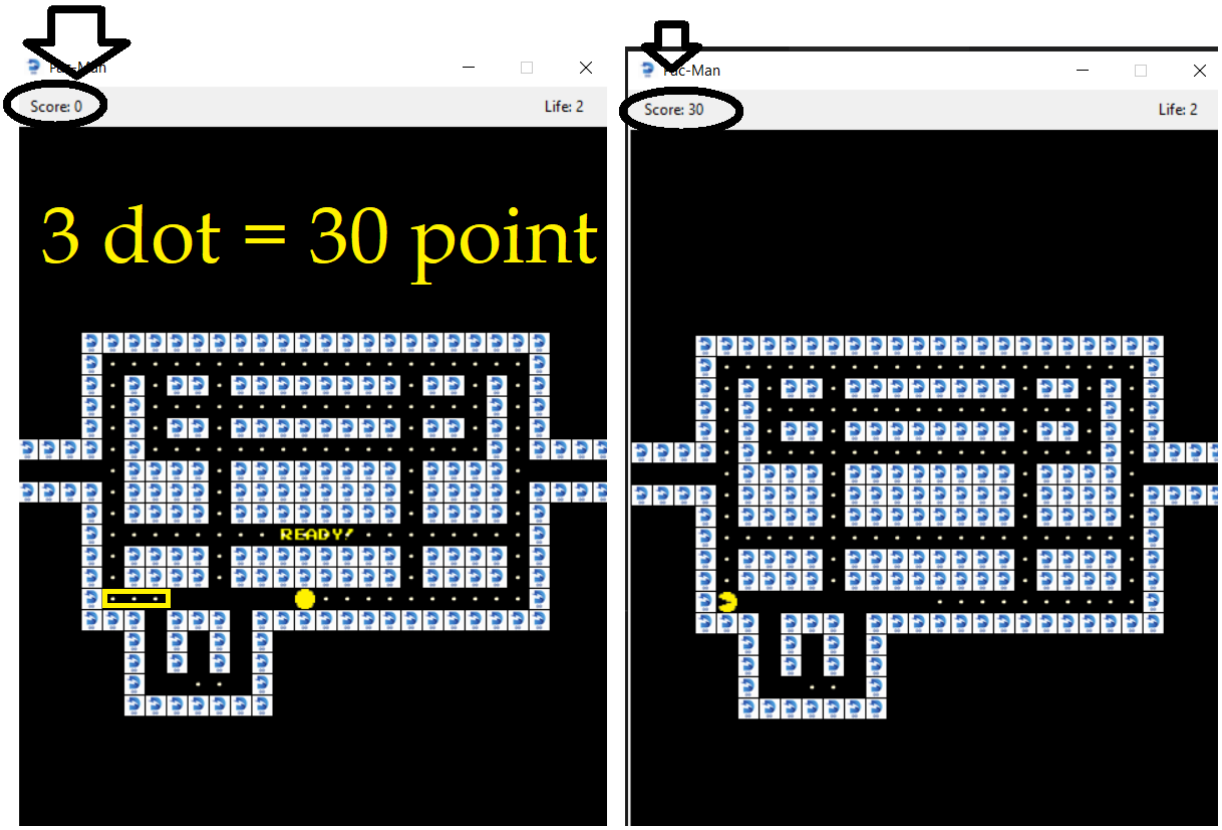
As levels are difficult to clear, so we are giving three life to complete it.

Black Box Testing→



3 User will get 10 point after eating each food

Black box testing→



4 User can skip demo video (in the beginning of game)

Black Box testing→

```
def __initLevelStarting(self):
    self.statusStartingTimer += 1   # countdown timer for this function

    # bind the sprite for the widget
    self.wGameCanv.itemconfig(self.wGameCanvLabelGetReady, image=self.wSprites['getready'])

    if self.statusStartingTimer < 16:
        # blinking function
        if self.statusStartingTimer % 2 == 1:
            self.wGameCanv.itemconfigure(self.wGameCanvLabelGetReady, state='normal')
        else:
            self.wGameCanv.itemconfigure(self.wGameCanvLabelGetReady, state='hidden')

    else:   # after 16 loop, the main game will be started with loopFunction
        self.gameStartingTrigger()


def gameStartingTrigger(self):
    ## stop to print out 'get ready' and start the game
    self.timerReady.stop()
    self.wGameCanv.itemconfigure(self.wGameCanvLabelGetReady, state='hidden')
    self.statusStartingTimer = 0
    self.isPlaying = True
```

5 User have to play by using arrow keys.

5.1 when user enter “Left” arrow

```
if field.gameEngine.movingObjectPacman.dirCurrent == "Left":

    # check the object passed field edges
    if field.gameEngine.movingObjectPacman.dirEdgePassed == True:
        self.wGameCanv.move(self.wGameCanvMovingObjects[0], 17*27+17, 0)    # notice this will mov
        field.gameEngine.movingObjectPacman.dirEdgePassed = False
    else:
        pass

    if coordAbsP[0] % 4 == 0:
        self.wGameCanv.itemconfig(self.wGameCanvMovingObjects[0], image=self.wSprites['pacmanL2'])
        self.wGameCanv.move(self.wGameCanvMovingObjects[0], -4, 0)
    elif coordAbsP[0] % 4 == 1:
        self.wGameCanv.itemconfig(self.wGameCanvMovingObjects[0], image=self.wSprites['pacmanL3'])
        self.wGameCanv.move(self.wGameCanvMovingObjects[0], -4, 0)
    elif coordAbsP[0] % 4 == 2:
        self.wGameCanv.itemconfig(self.wGameCanvMovingObjects[0], image=self.wSprites['pacmanL2'])
        self.wGameCanv.move(self.wGameCanvMovingObjects[0], -4, 0)
    elif coordAbsP[0] % 4 == 3:
        self.wGameCanv.itemconfig(self.wGameCanvMovingObjects[0], image=self.wSprites['pacmanL1'])
        self.wGameCanv.move(self.wGameCanvMovingObjects[0], -5, 0)
```

5.2 When User Enter “Right” arrow

```
elif field.gameEngine.movingObjectPacman.dirCurrent == "Right":

    # check the object passed field edges
    if field.gameEngine.movingObjectPacman.dirEdgePassed == True:
        self.wGameCanv.move(self.wGameCanvMovingObjects[0], -(17*27+17), 0)
        field.gameEngine.movingObjectPacman.dirEdgePassed = False
    else:
        pass

    if coordAbsP[0] % 4 == 0:
        self.wGameCanv.itemconfig(self.wGameCanvMovingObjects[0], image=self.wSprites['pacmanR2'])
        self.wGameCanv.move(self.wGameCanvMovingObjects[0], 4, 0)
    elif coordAbsP[0] % 4 == 1:
        self.wGameCanv.itemconfig(self.wGameCanvMovingObjects[0], image=self.wSprites['pacmanR3'])
        self.wGameCanv.move(self.wGameCanvMovingObjects[0], 4, 0)
    elif coordAbsP[0] % 4 == 2:
        self.wGameCanv.itemconfig(self.wGameCanvMovingObjects[0], image=self.wSprites['pacmanR2'])
        self.wGameCanv.move(self.wGameCanvMovingObjects[0], 4, 0)
    elif coordAbsP[0] % 4 == 3:
        self.wGameCanv.itemconfig(self.wGameCanvMovingObjects[0], image=self.wSprites['pacmanR1'])
        self.wGameCanv.move(self.wGameCanvMovingObjects[0], 5, 0)
```

5.3 when user enter “Up” arrow

```
    elif field.gameEngine.movingObjectPacman.dirCurrent == "Up":

        # check the object passed field edges
        if field.gameEngine.movingObjectPacman.dirEdgePassed == True:
            self.wGameCanv.move(self.wGameCanvMovingObjects[0], 0, 17*31+17)
            field.gameEngine.movingObjectPacman.dirEdgePassed = False
        else:
            pass

        if coordAbsP[1] % 4 == 0:
            self.wGameCanv.itemconfig(self.wGameCanvMovingObjects[0], image=self.wSprites['pacmanU2'])
            self.wGameCanv.move(self.wGameCanvMovingObjects[0], 0, -4)
        elif coordAbsP[1] % 4 == 1:
            self.wGameCanv.itemconfig(self.wGameCanvMovingObjects[0], image=self.wSprites['pacmanU3'])
            self.wGameCanv.move(self.wGameCanvMovingObjects[0], 0, -4)
        elif coordAbsP[1] % 4 == 2:
            self.wGameCanv.itemconfig(self.wGameCanvMovingObjects[0], image=self.wSprites['pacmanU2'])
            self.wGameCanv.move(self.wGameCanvMovingObjects[0], 0, -4)
        elif coordAbsP[1] % 4 == 3:
            self.wGameCanv.itemconfig(self.wGameCanvMovingObjects[0], image=self.wSprites['pacmanU1'])
            self.wGameCanv.move(self.wGameCanvMovingObjects[0], 0, -5)
```

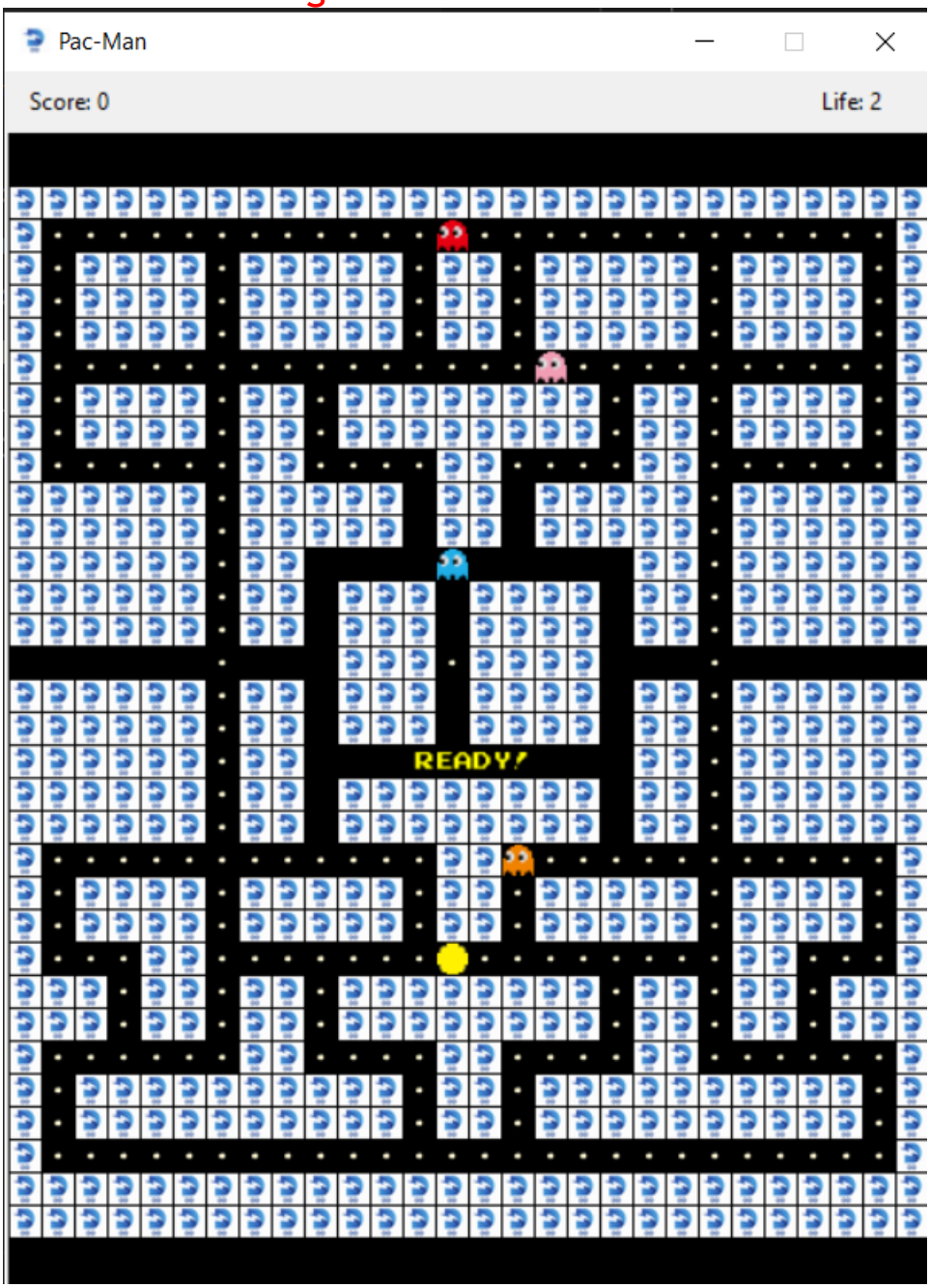
5.4 When User Enter “Down” arrow

```
elif field.gameEngine.movingObjectPacman.dirCurrent == "Down":

    # check the object passed field edges
    if field.gameEngine.movingObjectPacman.dirEdgePassed == True:
        self.wGameCanv.move(self.wGameCanvMovingObjects[0], 0, -(17*31+17))
        field.gameEngine.movingObjectPacman.dirEdgePassed = False
    else:
        pass

    if coordAbsP[1] % 4 == 0:
        self.wGameCanv.itemconfig(self.wGameCanvMovingObjects[0], image=self.wSprites['pacmanD2'])
        self.wGameCanv.move(self.wGameCanvMovingObjects[0], 0, 4)
    elif coordAbsP[1] % 4 == 1:
        self.wGameCanv.itemconfig(self.wGameCanvMovingObjects[0], image=self.wSprites['pacmanD3'])
        self.wGameCanv.move(self.wGameCanvMovingObjects[0], 0, 4)
    elif coordAbsP[1] % 4 == 2:
        self.wGameCanv.itemconfig(self.wGameCanvMovingObjects[0], image=self.wSprites['pacmanD2'])
        self.wGameCanv.move(self.wGameCanvMovingObjects[0], 0, 4)
    elif coordAbsP[1] % 4 == 3:
        self.wGameCanv.itemconfig(self.wGameCanvMovingObjects[0], image=self.wSprites['pacmanD1'])
        self.wGameCanv.move(self.wGameCanvMovingObjects[0], 0, 5)
```

6 Game contains Images

There are images of wall, food, Pac-Man, different images of all ghosts.

Black box Testing→



7 Game contains sound effects

For better experience we are using sound effects

```
self.wSounds = {
    'chomp1': pygame.mixer.Sound(file="resource/sound_chomp1.wav"),
    'chomp2': pygame.mixer.Sound(file="resource/sound_chomp2.wav")
}

# advance to next phase: get ready!
pygame.mixer.music.stop()
pygame.mixer.music.load("resource/sound_intro.mp3")
# ghost sound as music
pygame.mixer.music.stop()
pygame.mixer.music.load("resource/sound_ghost.ogg")
pygame.mixer.music.play(-1)

# sound effect
pygame.mixer.music.load("resource/sound_death.mp3"
pygame.mixer.music.play(loops=0, start=0.0)
```

8 User can End the Game whenever he wants

Press Esc to exit the game.

```
def inputResponseEsc(self, event):
    self.timerLoop.stop()
    messagebox.showinfo("Game Over!", "You hit the escape key!")
```