Group Members:

Zubaria Ayub (17i-0231)

Mishaal Naeem (17i-0326)

Project Report:

* Uses 3 macros:
  + Castle.inc
  + Flag.inc
  + Flower.inc
* StartPage:

Shows instructions to play the game through printText proc

* drawFlower:

draws flower using rectangle. Flower changes mario’s colour for 30 incs and allows advantage over enemies.

* checkCoinCollission

checks if Mario collides with a coin, adds score and removes coin

* checkFlowerCollission

checks if Mario collides with the flower, gives Mario immunity for 30 cycles

* NegativeCollission

Called when Mario collides with an object that reduces its life, reduces score by 25 and respawns Mario. If score <= 0, Game is Over, You Lose

* monsterMovement

moves the monster on top of screen and its thrown objects

* ChangeBackground:

Changes background colour to purple.

* printRectangle

prints a rectangle of height x width at coordinates (row,column) as the top left anchor point of mentioned colored. These come as arguments via stack

* drawEnemy

uses the printEnemy array to print the enemy on the screen

* drawMario

calls printRectangle multiple times to print the body of Mario using the top left as anchor point which is also (marioRow,marioCol). Takes an input in bx. If bx==0, then it prints all rectangles of Background color, else draws using the original colors. It also caters to the color change effect due to the flower

* printMario

passes 1 in bx and calls drawMario

* clearMario

passes 0 in bx and calls drawMario

* drawHurdle:

uses arrays with hurdle cords to dray a hurdle

* drawHurdles:

calls above proc for 3 different arrays

* enemyMovement

clears the previous image of enemy, changes its anchor column and prints it at the new coordinates. Ensures that the enemies stay between their own hurdles

* updateLandAndCheckEnemyCollission

checks the row after which gravity doesn’t affect Mario by dividing screen into 7 parts.

Part 1: between starting point and hurdle 1

Part 2: on top of hurdle 1, also checks for coin collission

Part 3: between hurdle 1 and hurdle 2, also checks for collissions with enemy1

Part 4: on top of hurdle 2, also checks for coin collision

Part 5: between hurdle 2 and hurdle 3, also checks for collision with enemy2

Part 6: on top of hurdle 3, also checks for coin collision

Part 7: between hurdle 3 and flag

* checkEnemyCollission

takes as input row and column of enemy and checks if Mario is colliding with it from right, left, top

* drawMonster:

uses array to draw a monster in level 3

* drawCoin:

uses array and rectangles to draw a coin of given array

* drawCoins:

calls above from for 3 different coin arrays

* PrintText:

Prints string

* PrintLevel:

Uses printText to display level in top-centre of screen

* printIntString:

converts int to string and prints it

* PrintScore:

Uses above proc to print score on top left

* EndScreen:

Displays end screen of game, and score of user

* InputUserName

Asks users to input name before starting game

Utitlity:

* NextLine
* timer
* clearSc

Screenshots: