LEVEL ONE (30*17)

```
#include <iostream>
#include <conio.h>
#include <stdio.h>
using namespace std;

void Background1(char X[][130])
{
                        for (int r = 0; r < 24; r++)
                        {
                          for (int c = 0; c < 75; c++)
                          {
                                X[r][c] = ' ';
                          }
                        }

                        for (int r = 0; r < 12; r++)
                        {
                          for (int c = 75; c < 80; c++)
                          {
                                X[r][c] = ' ';
                          }
                        }

                        //big starting block
                        for (int r = 20; r < 24; r++)
                        {
                          for (int c = 0; c < 7; c++)
                          {
                                X[r][c] = 178;
                          }
```

```
}

//path blocks
for (int c = 7; c < 80; c++)
{
  X[23][c] = 178;
}

//spikes
for (int c = 14; c < 18; c++)
{
  X[23][c] = 202;
}

//2 ladders with a surface at the top
for (int r = 17; r < 23; r++)
{
  X[r][24] = '|';
  X[r][25] = '_';
  X[r][26] = '|';
}
for (int c = 24; c < 44; c++)
{
  X[16][c] = 223;
}
for (int r = 17; r < 23; r++)
{
  X[r][41] = '|';
  X[r][42] = '_';
  X[r][43] = '|';
}
```

```
                        //collectable item
                        X[15][39] = 21;



                        //end-level door
                        for (int r = 13; r < 23; r++)
                        {
                          X[r][75] = 186;
                        }
                        X[12][75] = 201;
                        for (int c = 76; c < 80; c++)
                        {
                          X[12][c] = 205;
                        }

}

//coins
void coinsX(char X[][130], int& flagcoin1, int& flagcoin2, int& flagcoin3)
{
                        if (flagcoin1 == 1)
                        {
                          X[22][47] = ' ';
                          X[9][9] = '1';
                        }
                        else
                        {
                          X[22][47] = 233;
                        }

                        if (flagcoin2 == 1)
```

```
                        {
                          X[22][49] = ' ';
                          X[9][9] = '2';
                        }
                        else
                        {
                          X[22][49] = 233;
                        }

                        if (flagcoin3 == 1)
                        {
                          X[22][51] = ' ';
                          X[9][9] = '3';
                        }
                        else
                        {
                          X[22][51] = 233;
                        }
                        X[9][2] = 'c';
                        X[9][3] = 'o';
                        X[9][4] = 'i';
                        X[9][5] = 'n';
                        X[9][6] = 's';
                        X[9][7] = ':';
                        X[9][8] = ' ';
}

void itemX(char X[][130], int& checkitem)
{
                        if (checkitem == 0)
                        {
                          X[15][39] = 21;
```

```
                        }
                        else
                        {
                          X[15][39] = ' ';
                        }

}

void DrawArrowsX(char X[][130], int r, int c)
{
                        X[r][c] = 239;
                        X[r + 1][c] = '|';
                        X[r + 2][c] = ' ';

}

void MoveArrowX(int& r, int& c, int& dirarrow)
{
                        if (dirarrow == 1)
                        {
                          r++;
                        }
                        if (dirarrow == -1)
                        {
                          r--;
                        }
                        if (r == 23)
                        {
                          dirarrow = -1;
                        }
                        if (r == 21)
                        {
```

```
                        dirarrow = 1;
                    }


}


void HarakHeroX(char X[][130], int& rHero, int& cHero, char mv, int& flagHeroJump,
int& flagHeroLadder1, int& flagHeroLadder2, int& ctj, int& ctl1, int& ctl2, int&
surfaceCheck, int& flagHeroEdge, int& cted)
{
                    if (mv == 'w' && X[rHero - 1][cHero] != ' ')
                    {
                        rHero--;


                    }


                    if (mv == 's' && X[rHero + 3][cHero] == ' ' || cHero == 42)
                    {
                        if (rHero < 20)
                        {
                            rHero++;
                        }


                    }



                    if (mv == 'a')
                    {
                        if (cHero > 1)
                        {
                            cHero--;
                        }


                    }
```

```
if (mv == 'd')
{
   if (cHero < 80)
   {
          cHero++;
   }
}

if (flagHeroJump == 1)
{
   if (ctj == 0)
   {
          rHero--;
          cHero++;
   }
   if (ctj == 1)
   {
          cHero++;
   }
   if (ctj == 2)
   {
          cHero++;
   }
   if (ctj == 3)
   {
          cHero++;
   }
   if (ctj == 4)
   {
          cHero++;
   }
```

```
        if (ctj == 5)
        {
              cHero++;
        }
        if (ctj == 6)
        {
              rHero++;
              cHero++;
        }
        ctj++;
        if (ctj > 6 && surfaceCheck == 0)
        {
              if (rHero < 20)
              {
                    rHero++;
              }
              else
              {
                    flagHeroJump = 0;
              }
        }

}

if (flagHeroLadder1 == 1)
{
        if (ctl1 == 0)
        {
              rHero--;
        }
        if (ctl1 == 1)
        {
```

```
            rHero--;
        }
        if (ctl1 == 2)
        {
            rHero--;
        }
        if (ctl1 == 3)
        {
            rHero--;
            flagHeroLadder1 = 0;
        }
        ctl1++;

    }

    if (flagHeroLadder2 == 1)
    {
        if (ctl2 == 0)
        {
            rHero++;
        }
        if (ctl2 == 1)
        {
            rHero++;
        }

        if (ctl2 == 2)
        {
            rHero++;
            flagHeroLadder2 = 0;
        }
        ctl2++;
```

```
                        }

                        if (flagHeroEdge == 1)
                        {
                           if (cted == 0)
                           {
                                  rHero++;
                           }
                           if (cted == 1)
                           {
                                  rHero++;
                           }
                           if (cted == 2)
                           {
                                  rHero++;
                                  flagHeroEdge = 0;
                           }
                           cted++;
                        }

}


void ErsemHeroX(char X[][130], int rHero, int cHero)
{
                        X[rHero][cHero] = 1;
                        X[rHero + 1][cHero] = 30;
                        X[rHero + 1][cHero - 1] = '/';
                        X[rHero + 1][cHero + 1] = '\\';
                        X[rHero + 2][cHero - 1] = '/';
                        X[rHero + 2][cHero + 1] = '\\';
```

```
}

void ErsemEnemy1(char X[][130], int rEnemy1, int cEnemy1)
{
                        X[rEnemy1][cEnemy1] = 153;
                        X[rEnemy1 + 1][cEnemy1 - 1] = 185;
                        X[rEnemy1 + 1][cEnemy1 - 3] = 17;
                        X[rEnemy1 + 1][cEnemy1 + 1] = 185;
                        X[rEnemy1 + 2][cEnemy1] = 203;



}

void ErsemEnemy2(char X[][130], int rEnemy2, int cEnemy2)
{
                        X[rEnemy2][cEnemy2] = 232;
                        X[rEnemy2 + 1][cEnemy2 - 1] = 201;
                        X[rEnemy2 + 1][cEnemy2] = 223;
                        X[rEnemy2 + 1][cEnemy2 + 1] = 187;
}
void HarakEnemy2(char X[][130], int& i)
{
                        if (i <= 8)
                        {
                           int r = 13 + i;
                           X[r][76] = 220;
                           X[r][77] = 220;
                           X[r][78] = 220;
                           X[r][79] = 220;
                           i++;
                        }
```

```
}

void DisplayX(char X[][130], int& smvc)
{
                        system("cls");
                        int c;
                        if (smvc - 5 >= 0 && smvc + 25 < 80)
                        {
                          for (int r = 7; r < 24; r++)
                          {
                                for (c = smvc - 5; c < 25 + smvc; c++)
                                {
                                      cout << X[r][c];
                                }
                          }
                        }
                        else if (smvc + 25 >= 80)
                        {
                          for (int r = 7; r < 24; r++)
                          {
                                for (c = 50; c < 80; c++)
                                {
                                      cout << X[r][c];
                                }
                          }
                        }
                        else
                        {
                          for (int r = 7; r < 24; r++)
                          {
                                for (c = 0; c < 30; c++)
                                {
```

```
                                        cout << X[r][c];
                                }
                        }
                }

}


void main()
{
                int flagHeroJumpX = 0, ctj = 0, surfaceCheck = 0;
                char ch;
                int rarrow1x = 21, carrow1x = 59, dirarrow1x = 1;
                int rarrow2x = 21, carrow2x = 60, dirarrow2x = 1;
                int rarrow3x = 21, carrow3x = 61, dirarrow3x = 1;
                int rHerox = 17, cHerox = 3;
                int rEnemy1 = 13, cEnemy1 = 37;
                int rEnemy2 = 10, cEnemy2 = 77, e2startc = 76, e2endc = 79,
emy2r = 13, i = 0, k = 0;
                int doneladder = 0;
                int flagcoin1 = 0, flagcoin2 = 0, flagcoin3 = 0;
                int flagHeroLadder1 = 0, flagHeroLadder2 = 0;
                int ctl1 = 0, ctl2 = 0, SmasherCheck = 0, cts = 0, sbc = 31,
stc = 31;
                int checkitemx = 0, flagHeroEdge = 0, cted = 0, flagEnemy1 =
0, checkwin = 0, checklost = 0;
                int smvc = 0, phc = 1, phr = 0;
                char X[24][130];

                for (int r = 13; r < 23; r++)
                {
                  for (int c = 76; c < 80; c++)
                  {
                        X[r][c] = ' ';
```

```
   }
}
for (int r = 0; r < 24; r++)
{
   for (int c = 80; c < 130; c++)
   {
          X[r][c] = 178;
   }
}
//LEVEL ONE

for (;;)
{
   for (; !_kbhit();)
   {
          Background1(X);
          coinsX(X, flagcoin1, flagcoin2, flagcoin3);
          itemX(X, checkitemx);
          MoveArrowX(rarrow1x, carrow1x, dirarrow1x);
          DrawArrowsX(X, rarrow1x, carrow1x);
          MoveArrowX(rarrow2x, carrow2x, dirarrow2x);
          DrawArrowsX(X, rarrow2x, carrow2x);
          MoveArrowX(rarrow3x, carrow3x, dirarrow3x);
          DrawArrowsX(X, rarrow3x, carrow3x);
          ErsemHeroX(X, rHerox, cHerox);
          if (flagEnemy1 == 0)
          {
                 ErsemEnemy1(X, rEnemy1, cEnemy1);
          }

          if (cHerox >= 50 && k % 12 == 0)
          {
```

```
                                        HarakEnemy2(X, i);
                                }
                                k++;


                                ErsemEnemy2(X, rEnemy2, cEnemy2);
                                if (flagHeroJumpX == 1)
                                {
                                        if (X[rHerox + 2][cHerox] != ' ')
                                        {
                                                surfaceCheck = 1;
                                        }
                                        HarakHeroX(X, rHerox, cHerox, ch,
flagHeroJumpX, flagHeroLadder1, flagHeroLadder2, ctj, ctl1, ctl2, surfaceCheck,
flagHeroEdge, cted);
                                }


                                if (rHerox == 20)
                                {
                                        if (ctj > 6)
                                        {
                                                flagHeroJumpX = 0;
                                                surfaceCheck = 0;
                                                ctj = 0;
                                        }
                                }


                                if (flagHeroLadder1 == 1)
                                {
                                        HarakHeroX(X, rHerox, cHerox, ch,
flagHeroJumpX, flagHeroLadder1, flagHeroLadder2, ctj, ctl1, ctl2, surfaceCheck,
flagHeroEdge, cted);
                                }
                                if (flagHeroLadder2 == 1)
                                {
```

```
                                        HarakHeroX(X, rHerox, cHerox, ch,
flagHeroJumpX, flagHeroLadder1, flagHeroLadder2, ctj, ctl1, ctl2, surfaceCheck,
flagHeroEdge, cted);
                                }
                                if (flagHeroEdge == 1)
                                {
                                        HarakHeroX(X, rHerox, cHerox, ch,
flagHeroJumpX, flagHeroLadder1, flagHeroLadder2, ctj, ctl1, ctl2, surfaceCheck,
flagHeroEdge, cted);
                                }

                                //coins
                                if (rHerox == 20 && cHerox == 47)
                                {
                                        flagcoin1 = 1;
                                        X[7][77] = '1';
                                }
                                if (rHerox == 20 && cHerox == 49)
                                {
                                        flagcoin2 = 1;
                                        X[7][77] = '2';
                                }
                                if (rHerox == 20 && cHerox == 51)
                                {
                                        flagcoin3 = 1;
                                        X[7][77] = '3';
                                }

                                if (checkitemx == 0 && rHerox == 13 && cHerox ==
37)
                                {
                                        checkitemx++;

                                }
```

```
if (checkitemx > 0 && cHerox < 50)
{
	X[10][smvc - 11] = 'P';
	X[10][smvc - 10] = 'E';
	X[10][smvc - 9] = 'R';
	X[10][smvc - 8] = 'F';
	X[10][smvc - 7] = 'E';
	X[10][smvc - 6] = 'C';
	X[10][smvc - 5] = 'T';
	X[10][smvc - 4] = '!';
	X[10][smvc - 3] = ' ';
	X[10][smvc - 2] = 'i';
	X[10][smvc - 1] = 't';
	X[10][smvc] = 'e';
	X[10][smvc + 1] = 'm';
	X[10][smvc + 2] = ' ';
	X[10][smvc + 3] = 'c';
	X[10][smvc + 4] = 'o';
	X[10][smvc + 5] = 'l';
	X[10][smvc + 6] = 'l';
	X[10][smvc + 7] = 'e';
	X[10][smvc + 8] = 'c';
	X[10][smvc + 9] = 't';
	X[10][smvc + 10] = 'e';
	X[10][smvc + 11] = 'd';

}



//touch first spikes
```

```
                                if (rHerox + 2 == 22 && cHerox + 1 == 14 || rHerox
+ 2 == 22 && cHerox + 1 == 15 || rHerox + 2 == 22 && cHerox + 1 == 16 || rHerox + 2
== 22 && cHerox + 1 == 17 || rHerox + 2 == 22 && cHerox + 1 == 18)
                                {
                                      rHerox = 17;
                                      cHerox = 3;
                                      flagHeroEdge = 0;
                                      cted = 0;
                                      X[9][smvc] = 'R';
                                      X[9][smvc + 1] = 'E';
                                      X[9][smvc + 2] = 'T';
                                      X[9][smvc + 3] = 'R';
                                      X[9][smvc + 4] = 'Y';

                                }

                                //touch moving arrows
                                if (rHerox + 2 == 22 && cHerox + 1 == 59 || rHerox
+ 2 == 22 && cHerox + 1 == 60 || rHerox + 2 == 22 && cHerox + 1 == 61 || rHerox + 2
== 22 && cHerox - 1 == 59 || rHerox + 2 == 22 && cHerox - 1 == 60 || rHerox + 2 ==
22 && cHerox - 1 == 61)
                                {
                                      rHerox = 17;
                                      cHerox = 3;
                                      flagHeroEdge = 0;
                                      cted = 0;
                                      X[9][smvc] = 'R';
                                      X[9][smvc + 1] = 'E';
                                      X[9][smvc + 2] = 'T';
                                      X[9][smvc + 3] = 'R';
                                      X[9][smvc + 4] = 'Y';

                                }
```

```
                                    //touch enemy weapon
                                    if (rHerox == 13 && cHerox > 29 && SmasherCheck ==
0 && cHerox + 1 == 34)
                                    {
                                          rHerox = 17;
                                          cHerox = 3;
                                          flagHeroEdge = 0;
                                          cted = 0;
                                          X[9][smvc] = 'R';
                                          X[9][smvc + 1] = 'E';
                                          X[9][smvc + 2] = 'T';
                                          X[9][smvc + 3] = 'R';
                                          X[9][smvc + 4] = 'Y';

                                    }

                                    if (cHerox == 28 && rHerox == 13)
                                    {
                                          X[13][stc] = 16;
                                          X[14][sbc] = 186;
                                    }

                                    if (cHerox == 29 && SmasherCheck == 1)
                                    {
                                          X[14][stc + 3] = 16;
                                          X[14][sbc + 2] = 205;
                                          flagEnemy1 = 1;
                                          X[8][smvc + 2] = 'E';
                                          X[8][smvc + 3] = 'N';
                                          X[8][smvc + 4] = 'E';
                                          X[8][smvc + 5] = 'M';
                                          X[8][smvc + 6] = 'Y';
                                          X[8][smvc + 7] = ' ';
```

```
                                        X[8][smvc + 8] = 'V';
                                        X[8][smvc + 9] = 'A';
                                        X[8][smvc + 10] = 'N';
                                        X[8][smvc + 11] = 'Q';
                                        X[8][smvc + 12] = 'U';
                                        X[8][smvc + 13] = 'I';
                                        X[8][smvc + 14] = 'S';
                                        X[8][smvc + 15] = 'H';
                                        X[8][smvc + 16] = 'E';
                                        X[8][smvc + 17] = 'D';

                                }
                                if (smvc <= 78)
                                {
                                        if (phc < cHerox)
                                        {
                                                smvc = smvc + (cHerox - phc);
                                        }
                                        else
                                        {
                                                smvc = smvc - (phc - cHerox);
                                        }
                                }

                                DisplayX(X, smvc);

                                phc = cHerox;
                        }

                        ch = _getch();
                        HarakHeroX(X, rHerox, cHerox, ch, flagHeroJumpX,
flagHeroLadder1, flagHeroLadder2, ctj, ctl1, ctl2, surfaceCheck, flagHeroEdge,
cted);
```

```
if (ch == ' ')
{
	flagHeroJumpX = 1;
}
if (ch == 'w' && rHerox == 19 && cHerox == 25)
{
	flagHeroLadder1 = 1;
}
if (ch == 's' && rHerox == 14 && cHerox == 42)
{
	flagHeroLadder2 = 1;
}
if (ch == 'd' && cHerox == 7)
{
	flagHeroEdge = 1;
}
ErsemHeroX(X, rHerox, cHerox);


if (ch == 'h')
{
	SmasherCheck = 1;

}


//lose or win

//case1:door closed before passing with/without item
if (X[19][76] != ' ')
{
	for (int r = 0; r < 24; r++)
```

```
		{
			for (int c = 0; c < 130; c++)
			{
				X[r][c] = ' ';
			}
		}

		X[13][smvc - 5] = 'O';
		X[13][smvc - 4] = 'P';
		X[13][smvc - 3] = 'P';
		X[13][smvc - 2] = 'S';
		X[13][smvc - 1] = '.';
		X[13][smvc] = '.';
		X[13][smvc + 1] = '.';
		X[14][smvc - 5] = 'D';
		X[14][smvc - 4] = 'O';
		X[14][smvc - 3] = 'O';
		X[14][smvc - 2] = 'R';
		X[15][smvc - 5] = 'B';
		X[15][smvc - 4] = 'L';
		X[15][smvc - 3] = 'O';
		X[15][smvc - 2] = 'C';
		X[15][smvc - 1] = 'K';
		X[15][smvc] = 'E';
		X[15][smvc + 1] = 'D';
		checklost = 1;

	}

	//case2:passed through door with no item
	else if (checkitemx == 0 && cHerox >= 75)
	{
```

```
for (int r = 0; r < 24; r++)
{
	for (int c = 0; c < 130; c++)
	{
		X[r][c] = ' ';
	}
}

X[13][smvc - 6] = 'N';
X[13][smvc - 5] = 'O';
X[13][smvc - 4] = ' ';
X[13][smvc - 3] = 'I';
X[13][smvc - 2] = 'T';
X[13][smvc - 1] = 'E';
X[13][smvc] = 'M';
X[13][smvc + 1] = ' ';
X[13][smvc + 2] = ':';
X[13][smvc + 3] = '(';
X[15][smvc - 10] = 'I';
X[15][smvc - 9] = 'T';
X[15][smvc - 8] = ' ';
X[15][smvc - 7] = 'I';
X[15][smvc - 6] = 'S';
X[15][smvc - 5] = ' ';
X[15][smvc - 4] = 'A';
X[15][smvc - 3] = ' ';
X[15][smvc - 2] = 'M';
X[15][smvc - 1] = 'U';
X[15][smvc] = 'S';
X[15][smvc + 1] = 'T';
X[15][smvc + 2] = '.';
X[15][smvc + 3] = '.';
```

```
			X[15][smvc + 4] = '.';
			checklost = 1;

		}
		else if (checkitemx == 1 && cHerox >= 75)
		{
			for (int r = 0; r < 24; r++)
			{
				for (int c = 0; c < 130; c++)
				{
					X[r][c] = ' ';
				}
			}
			X[16][smvc - 6] = 'Y';
			X[16][smvc - 5] = 'A';
			X[16][smvc - 4] = 'Y';
			X[16][smvc - 3] = ' ';
			X[16][smvc - 2] = 'Y';
			X[16][smvc - 1] = 'O';
			X[16][smvc] = 'U';
			X[16][smvc + 1] = ' ';
			X[16][smvc + 2] = 'W';
			X[16][smvc + 3] = 'O';
			X[16][smvc + 4] = 'N';
			X[16][smvc + 5] = '!';
			X[16][smvc + 6] = '!';
			checkwin = 1;
		}

		if (checkwin == 1 || checklost == 1)
		{
			DisplayX(X, smvc);
```

```
                                    break;
                                }


                            }

}
```

LEVEL TWO

```
#include <iostream>
#include <conio.h>
using namespace std;


void Background2(char y[][160])
{
   for (int r = 0; r < 19; r++)
   {
      for (int c = 0; c < 160; c++)
      {
         y[r][c] = ' ';
      }
   }

   for (int r = 20; r < 24; r++)
   {
      for (int c = 0; c < 160; c++)
      {
```

```
            y[r][c] = ' ';
        }
    }

    for (int c = 0; c < 65; c++)
    {
        y[19][c] = ' ';
    }
    for (int c = 80; c < 160; c++)
    {
        y[19][c] = ' ';
    }

    for (int r = 0; r < 24; r++)
    {
        y[r][0] = 179;
        y[r][159] = 179;
    }

    //starting blocks
    for (int c = 0; c < 160; c++)
    {
        y[23][c] = 178;
    }

    for (int r = 22; r < 24; r++)
    {
        for (int c = 5; c < 10; c++)
        {
            y[r][c] = 178;

        }
    }

    for (int r = 21; r < 24; r++)
    {
        for (int c = 9; c < 15; c++)
        {
            y[r][c] = 178;
        }
    }


    //sky
    for (int r = 0; r < 3; r++)
    {
        for (int c = 0; c < 160; c++)
        {
```

```
            y[r][c] = 178;
        }
    }
    for (int r = 0; r < 6; r++)
    {
        for (int c = 16; c < 21; c++)
        {
            y[r][c] = 178;
        }
    }

    for (int r = 0; r < 16; r++)
    {
        for (int c = 45; c < 50; c++)
        {
            y[r][c] = 178;
        }
    }

    //bullets 1
    y[6][16] = 186;
    y[7][16] = 200;
    y[7][17] = 205;
    y[7][18] = 31;
    y[7][19] = 205;
    y[6][20] = 186;
    y[7][20] = 188;

    //bullets 2
    y[16][45] = 186;
    y[17][45] = 200;
    y[17][46] = 205;
    y[17][47] = 31;
    y[17][48] = 205;
    y[16][49] = 186;
    y[17][49] = 188;

    //tall block in middle
    for (int r = 22; r > 9; r--)
    {
        for (int c = 29; c < 34; c++)
        {
            y[r][c] = 178;
        }
    }

    //coins platforms
    //top
```

```
for (int c = 50; c < 84; c++)
{
   y[6][c] = 220;
}
//ladder 1
for (int r = 7; r < 11; r++)
{
   y[r][67] = 195;
   y[r][68] = '-';
   y[r][69] = 180;
}
//bottom
for (int c = 50; c < 84; c++)
{
   y[11][c] = 220;
}
for (int r = 3; r < 12; r++)
{
   y[r][83] = 219;
}
//ladder 2
for (int r = 12; r < 18; r++)
{
   y[r][78] = 195;
   y[r][79] = '-';
   y[r][80] = 180;
}


//first platforms
//no.1
y[21][56] = 220;
y[21][57] = 220;
y[21][58] = 220;
y[21][59] = 220;
y[21][60] = 220;
y[21][61] = 201;
y[21][62] = 254;
y[21][63] = 187;

//no.2 (before enemy)
y[19][65] = 220;
y[19][66] = 220;
y[19][67] = 220;
y[19][68] = 220;

//no.3
y[17][77] = 220;
```

```
y[17][78] = 220;
y[17][79] = 220;
y[17][80] = 220;
y[17][81] = 220;
y[17][82] = 220;
y[17][83] = 220;
y[17][84] = 220;
y[17][85] = 220;
y[17][86] = 220;
y[17][87] = 220;

//no.4
y[15][90] = 220;
y[15][91] = 220;
y[15][92] = 220;
y[15][93] = 220;
y[15][94] = 220;
y[15][95] = 220;

//no.5
y[13][98] = 220;
y[13][99] = 220;
y[13][100] = 220;
y[13][101] = 220;
y[13][102] = 220;
y[13][103] = 220;


//rope top-base
y[3][106] = 195;
for (int c = 107; c < 136; c++)
{
   y[3][c] = 194;
}
y[3][136] = 180;


//spikes
for (int c = 103; c < 136; c++)
{
   y[23][c] = 30;
}


//second platforms
//no.1
y[13][130] = 220;
y[13][131] = 220;
```

```
   y[13][132] = 220;
   y[13][133] = 220;
   y[13][134] = 220;
   y[13][135] = 220;
   y[13][136] = 220;
   y[13][137] = 220;


   //no.2
   y[15][138] = 220;
   y[15][139] = 220;
   y[15][140] = 220;
   y[15][141] = 220;
   y[15][142] = 220;
   y[15][143] = 220;
   y[15][144] = 220;
   y[15][145] = 220;

   //outside border of elevator
   y[10][145] = '.';
   for (int c = 146; c < 151; c++)
   {
      y[10][c] = '.';
   }
   y[10][151] = '.';

   for (int r = 11; r < 23; r++)
   {
      y[r][151] = ':';
   }
   for (int r = 11; r < 23; r++)
   {
      y[r][145] = ':';
   }
}

void endDoor(char y[][160], int& openDoor)
{
   if (openDoor == 0)
   {
      for (int r = 17; r < 23; r++)
      {
         y[r][155] = 186;
      }
      y[17][155] = 201;
      for (int c = 156; c < 159; c++)
      {
         y[17][c] = 205;
```

```
      }
   }
   else
   {

      for (int r = 17; r < 19; r++)
      {
         y[r][155] = 186;
      }
      y[17][155] = 201;
      for (int c = 156; c < 159; c++)
      {
         y[17][c] = 205;
      }
   }
}


void jetback(char y[][160], int flagJetBack)
{
   if (flagJetBack == 0)
   {
      y[19][12] = '[';
      y[19][13] = ']';
      y[19][11] = 204;
      y[19][14] = 185;
   }
   else
   {
      y[19][12] = ' ';
      y[19][13] = ' ';
      y[19][11] = ' ';
      y[19][14] = ' ';

   }
}

void enemyplatform(char y[][160], int& flagTouch, int cenemy3, int cHero)
{
   if (flagTouch == 1)
   {
      y[19][cenemy3] = ' ';
   }

}

//coins
```

```
void coinsY(char y[][160], int& flagcoin1, int& flagcoin2, int& flagcoin3, int& flagcoin4, int& flagcoin5,
int& flagcoin6, int& flagcoin7, int& flagcoin8, int& flagcoin9, int& flagcoin10, int& ctcoins)
{
   if (flagcoin1 == 1)
   {
      y[14][25] = ' ';

   }
   else
   {
      y[14][25] = 233;
   }

   if (flagcoin2 == 1)
   {
      y[11][25] = ' ';
   }
   else
   {
      y[11][25] = 233;
   }

   if (flagcoin3 == 1)
   {
      y[16][38] = ' ';

   }
   else
   {
      y[16][38] = 233;
   }

   if (flagcoin3 == 1)
   {
      y[20][38] = ' ';

   }
   else
   {
      y[20][38] = 233;
   }

   if (flagcoin5 == 1)
   {
      y[5][55] = ' ';

   }
   else
```

```
{
   y[5][55] = 233;
}

if (flagcoin6 == 1)
{
   y[5][57] = ' ';

}
else
{
   y[5][57] = 233;
}

if (flagcoin7 == 1)
{
   y[5][59] = ' ';
}
else
{
   y[5][59] = 233;
}
if (flagcoin8 == 1)
{
   y[22][139] = ' ';
}
else
{
   y[22][139] = 233;
}
if (flagcoin9 == 1)
{
   y[22][140] = ' ';
}
else
{
   y[22][140] = 233;
}
if (flagcoin10 == 1)
{
   y[22][141] = ' ';
}
else
{
   y[22][141] = 233;
}

//count coins
```

```
    if (ctcoins == 1)
    {
       y[4][150] = '1';
    }
    if (ctcoins == 2)
    {
       y[4][150] = '2';
    }
    if (ctcoins == 3)
    {
       y[4][150] = '3';
    }
    if (ctcoins == 4)
    {
       y[4][150] = '4';
    }
    if (ctcoins == 5)
    {
       y[4][150] = '5';
    }
    if (ctcoins == 6)
    {
       y[4][150] = '6';
    }
    if (ctcoins == 7)
    {
       y[4][150] = '7';
    }
    if (ctcoins == 8)
    {
       y[4][150] = '8';
    }
    if (ctcoins == 9)
    {
       y[4][150] = '9';
    }
    if (ctcoins == 10)
    {
       y[4][150] = '1';
       y[4][151] = '0';
    }

    y[4][142] = 'c';
    y[4][143] = 'o';
    y[4][144] = 'i';
    y[4][145] = 'n';
    y[4][146] = 's';
    y[4][147] = ':';
```

```
    y[4][148] = ' ';
    y[6][142] = 't';
    y[6][143] = 'a';
    y[6][144] = 'r';
    y[6][145] = 'g';
    y[6][146] = 'e';
    y[6][147] = 't';
    y[6][148] = ' ';
    y[6][149] = 'i';
    y[6][150] = 's';

    y[6][152] = '1';
    y[6][153] = '0';
}

void DrawElevator(char Y[][160], int relevator, int prevre, int flagElevator, int openDoor, int& celevator)
{
    Y[relevator][celevator - 4] = 196;
    for (int r = relevator + 1; r < relevator + 4; r++)
    {
        //Y[r][146] = '|';
        Y[r][celevator] = '|';
    }
    Y[relevator][celevator] = 191;
    for (int c1 = 147; c1 < 150; c1++)
    {
        Y[relevator][c1] = 196;

    }
    Y[relevator + 4][celevator - 4] = 196;
    for (int c1 = 147; c1 < 150; c1++)
    {
        Y[relevator + 4][c1] = 196;

    }
    Y[relevator + 4][celevator] = 217;
}

void MoveElevator(int& r, int& c, int& dielevator, int& rHero, int& flagElevator)
{
    if (dielevator == 1)
    {
        r++;
    }
    if (flagElevator == 1 && dielevator == 1)
    {
        rHero++;
    }
```

```
   if (dielevator == -1)
   {
      r--;
   }

   if (flagElevator == 1 && dielevator == -1)
   {
      rHero--;
   }

   if (r == 18)
   {
      dielevator = -1;
   }
   if (r == 11)
   {
      dielevator = 1;
   }

}

void ErsemHeroY(char Y[][160], int rHero, int cHero, int flagJetBack, int flagHoldRope, int flagMoveRope)
{
   if (flagJetBack == 0 || flagJetBack == 2)
   {
      Y[rHero][cHero] = 1;
      Y[rHero + 1][cHero] = 30;
      Y[rHero + 1][cHero - 1] = '/';
      Y[rHero + 1][cHero + 1] = '\\';
      Y[rHero + 2][cHero - 1] = '/';
      Y[rHero + 2][cHero + 1] = '\\';
   }

   if (flagJetBack == 1)
   {
      Y[rHero][cHero] = 1;
      Y[rHero + 1][cHero] = 30;
      Y[rHero + 1][cHero - 1] = '/';
      Y[rHero + 1][cHero - 2] = 204;
      Y[rHero + 1][cHero + 1] = '\\';
      Y[rHero + 1][cHero + 2] = 185;
      Y[rHero + 2][cHero - 1] = '/';
      Y[rHero + 2][cHero + 1] = '\\';

   }
   if (flagHoldRope == 1 || flagMoveRope == 1)
   {
```

```
      Y[rHero][cHero] = 1;
      Y[rHero + 1][cHero] = 30;
      Y[rHero + 1][cHero - 1] = '/';
      Y[rHero][cHero + 1] = '//';
      Y[rHero + 1][cHero + 1] = ' ';
      Y[rHero + 2][cHero - 1] = '/';
      Y[rHero + 2][cHero + 1] = '\\';
   }

}

void HarakHeroY(char Y[][160], int& rHero, int& cHero, char mv, int& flagJetpack, int& flagLadder1up,
int& ctl1up, int& flagLadder1down, int& ctl1down, int& flagLadder2up, int& ctl2up, int&
flagLadder2down, int& ctl2down, int& flagHoldRope, int& cthr, int& flagHeroMoveRope, int& ctmr, int&
surfacecheck1, int& flagjump, int& ctj, int& surfacecheck, int& flagfall, int& flagElevator, int& dielevator)
{
   if (mv == 'w' && Y[rHero - 1][cHero] == ' ' && flagLadder1up == 0 && flagLadder2up == 0 &&
flagJetpack == 0)
   {
      rHero--;

   }

   if (mv == 'w' && Y[rHero - 1][cHero] == ' ' && flagLadder1up == 0 && flagLadder2up == 0 &&
flagJetpack == 1 || mv == 'w' && Y[rHero - 1][cHero] == -23 && flagLadder1up == 0 && flagLadder2up ==
0 && flagJetpack == 1)
   {
      rHero--;

   }


   if (mv == 's' && Y[rHero + 3][cHero] == ' ' && flagJetpack == 0)
   {
      if (rHero < 20)
      {
         rHero++;
      }

   }

   if (mv == 's' && Y[rHero + 3][cHero] == ' ' && flagJetpack == 1 || mv == 's' && Y[rHero + 3][cHero] == -
23 && flagJetpack == 1)
   {
      if (Y[rHero + 4][cHero] == ' ' || Y[rHero + 4][cHero] == -23)
      {
         rHero++;
      }
```

```
   }

   if (mv == 'a' && Y[rHero][cHero - 1] == ' ' && Y[rHero + 1][cHero - 2] == ' ' && Y[rHero + 2][cHero - 2] ==
' ' || mv == 'a' && cHero == 15 || mv == 'a' && cHero == 14 || mv == 'a' && cHero == 13 || mv == 'a' &&
cHero == 12 || mv == 'a' && cHero == 71 || mv == 'a' && cHero == 70 || mv == 'a' && cHero == 69 || mv
== 'a' && cHero == 68 || mv == 'a' && cHero == 79 || mv == 'a' && cHero == 80 || mv == 'a' && cHero
== 81 || mv == 'a' && cHero == 82 || mv == 'a' && cHero == 61 || mv == 'a' && cHero == 59 || mv == 'a'
&& cHero == 57 || mv == 'a' && cHero >= 143 && cHero <= 153) {

      if (cHero > 1) {
         cHero--;

      }
   }

   if (mv == 'a' && Y[rHero][cHero - 2] == ' ' && Y[rHero + 1][cHero - 3] == ' ' && Y[rHero + 2][cHero - 3] ==
' ' && flagJetpack == 1 || mv == 'a' && flagJetpack == 1 && cHero == 28 || mv == 'a' && flagJetpack == 1
&& cHero == 41 || mv == 'a' && rHero == 20 && cHero == 143 || mv == 'a' && rHero == 20 && cHero ==
142 || mv == 'a' && rHero == 20 && cHero == 141)
   {
      if (cHero > 1)
      {
         cHero--;
      }

   }

   if (mv == 'd' && Y[rHero][cHero + 1] == ' ' && Y[rHero + 1][cHero + 2] == ' ' && Y[rHero + 2][cHero + 2]
== ' ' || mv == 'd' && cHero == 9 || mv == 'd' && cHero == 10 || mv == 'd' && cHero == 11 || mv == 'd'
&& cHero == 12 || mv == 'd' && cHero == 76 || mv == 'd' && cHero == 77 || mv == 'd' && cHero == 78
|| mv == 'd' && cHero == 79 || mv == 'd' && cHero == 80 || mv == 'd' && cHero == 65 || mv == 'd' &&
cHero == 66 || mv == 'd' && cHero == 67 || mv == 'd' && cHero == 68 || mv == 'd' && cHero == 55 ||
mv == 'd' && cHero == 57 || mv == 'd' && cHero == 59 || mv == 'd' && cHero >= 143 && cHero < 153)
   {
      if (flagElevator == 0)
      {
         if (cHero < 160)
         {
            cHero++;

         }
      }
      else
      {
         if (cHero >= 146 && cHero <= 148)
         {
            cHero++;
```

```
        }

      }

   }
   if (mv == 'd' && Y[rHero][cHero + 2] == ' ' && Y[rHero + 1][cHero + 3] == ' ' && Y[rHero + 2][cHero + 3]
== ' ' && flagJetpack == 1 || mv == 'd' && flagJetpack == 1 && cHero == 22 || mv == 'd' && flagJetpack
== 1 && cHero == 35)
   {
      if (cHero < 160)
      {
         cHero++;
      }
   }

   if (flagElevator == 1)
   {
      if (dielevator == 1)
      {
         rHero++;
      }
      if (dielevator == -1)
      {
         rHero--;
      }

   }

   if (flagLadder1up == 1)
   {
      if (ctl1up == 0)
      {
         rHero--;
      }
      if (ctl1up == 1)
      {
         rHero--;
      }
      if (ctl1up == 2)
      {
         rHero--;
      }
      if (ctl1up == 4)
      {
         rHero--;

      }
      if (ctl1up == 5)
```

```
    {
      rHero--;
      flagLadder1up = 0;
    }
    ctl1up++;
    if (ctl1up >= 6)
    {
      ctl1up = 0;
    }
  }

  if (flagLadder1down == 1)
  {
    if (ctl1down == 0)
    {
      rHero++;
    }
    if (ctl1down == 1)
    {
      rHero++;
    }
    if (ctl1down == 2)
    {
      rHero++;
    }
    if (ctl1down == 4)
    {
      rHero++;

    }
    if (ctl1down == 5)
    {
      rHero++;
      flagLadder1down = 0;
    }
    ctl1down++;
    if (ctl1down >= 6)
    {
      ctl1down = 0;
    }
  }

  if (flagLadder2up == 1)
  {
    if (ctl2up == 0)
    {
      rHero--;
    }
```

```
    if (ctl2up == 1)
    {
       rHero--;
    }
    if (ctl2up == 2)
    {
       rHero--;

    }
    if (ctl2up == 3)
    {
       rHero--;

    }
    if (ctl2up == 4)
    {
       rHero--;

    }
    if (ctl2up == 5)
    {
       rHero--;
       flagLadder2up = 0;
    }
    ctl2up++;
    if (ctl2up >= 6)
    {
       ctl2up = 0;
    }
  }

  if (flagLadder2down == 1)
  {
    if (ctl2down == 0)
    {
       rHero++;
    }
    if (ctl2down == 1)
    {
       rHero++;
    }
    if (ctl2down == 2)
    {
       rHero++;

    }
    if (ctl2down == 3)
    {
```

```
            rHero++;

        }
        if (ctl2down == 4)
        {
            rHero++;

        }
        if (ctl2down == 5)
        {
            rHero++;
            flagLadder2down = 0;

        }
        ctl2down++;
        if (ctl2down >= 6)
        {
            ctl2down = 0;
        }

    }

    if (flagHoldRope == 1)
    {
        if (cthr == 0)
        {
            rHero--;

        }
        if (cthr == 1)
        {
            cHero++;
            flagHeroMoveRope = 1;
        }
        cthr++;
    }
    if (flagHeroMoveRope == 1)
    {
        if (ctmr == 0)
        {
            cHero++;

        }
        ctmr++;
    }

    if (flagjump == 1)
    {
```

```
      if (ctj == 0)
      {
         cHero++;
         rHero--;
      }
      if (ctj == 1)
      {
         cHero++;
         rHero--;
      }
      if (ctj == 2)
      {
         cHero++;
      }
      if (ctj == 3)
      {
         cHero++;
      }
      if (ctj == 4 && Y[rHero + 3][cHero + 2] == ' ')
      {
         cHero++;
         rHero++;
      }
      if (ctj == 5 && Y[rHero + 3][cHero + 2] == ' ')
      {
         rHero++;
      }
      ctj++;
      if (ctj == 6)
      {
         ctj = -1;
         flagjump = 0;
      }

   }
   if (Y[rHero + 3][cHero - 1] == ' ' && Y[rHero + 3][cHero + 1] == ' ' && flagjump == 0 && flagJetpack == 0
&& flagHeroMoveRope == 0 && flagElevator == 0)
   {
      flagfall = 1;
   }
   else
   {
      flagfall = 0;
   }
   if (flagfall == 1 && flagHeroMoveRope == 0)
   {
      if (rHero < 20)
      {
```

```
            rHero++;
        }
    }

}

void MoveRope(int& cmvr, int& prevcr, int& flagMoveRope, int& cHero, int& flagHeroMoveRope, int&
flagHoldRope)
{
    prevcr = cmvr;
    if (cmvr < 136 && flagMoveRope == 1)
    {
        cmvr++;
        cHero++;
    }
    else
    {
        flagMoveRope = 0;
        flagHeroMoveRope = 0;
        flagHoldRope = 0;
    }

}

void DrawRope(char Y[][160], int cmvr, int prevcr, int flagMoveRope)
{
    for (int r = 4; r < 10; r += 2)
    {
        Y[r][cmvr] = 186;

    }
    for (int r = 5; r < 11; r += 2)
    {
        Y[r][cmvr] = 'O';

    }
    if (flagMoveRope == 1)
    {
        for (int r = 4; r < 11; r++)
        {
            Y[r][prevcr] = ' ';

        }
    }

}

void DrawArrows(char X[][160], int r, int c)
```

```
{
   X[r + 1][c] = '|';
   X[r + 2][c] = 31;

}

void MoveArrow(int& r, int& c, int& dirarrow)
{
   if (dirarrow == 1)
   {
      r++;
   }
   if (dirarrow == -1)
   {
      r--;
   }
   if (r == 2)
   {
      dirarrow = -1;
   }
   if (r == 1)
   {
      dirarrow = 1;
   }

}

void drawlaser(char X[][160], int r, int c)
{
   X[r + 1][c] = 179;
   X[r + 2][c] = 179;
}

void movelaser(int& r, int& c, int& dilaser)
{
   if (dilaser == 1)
   {
      r++;
   }
   if (dilaser == -1)
   {
      r--;
   }
   if (r == 2)
   {
      dilaser = -1;
   }
   if (r == 1)
```

```
    {
        dilaser = 1;
    }
}

void ErsemEnemy3(char Y[][160], int renemy3, int cenemy3)
{
    Y[renemy3][cenemy3] = 254;
    Y[renemy3 + 1][cenemy3] = 186;
}

void HarakEnemy3(char Y[][160], int& renemy3, int& cenemy3, int& direnemy, int& flagTouch)
{
    if (cenemy3 < 79)
    {
        if (direnemy == 1)
        {
            renemy3--;
        }
        if (direnemy == -1)
        {
            renemy3++;
        }
        if (renemy3 == 19)
        {
            direnemy = -1;
        }
        if (renemy3 == 20)
        {
            direnemy = 1;
        }

        cenemy3++;
    }
    if (renemy3 == 19)
    {
        flagTouch = 1;
    }

}

void DisplayY(char Y[][160])
{
    system("cls");
    for (int r = 0; r < 24; r++)
    {
        for (int c = 0; c < 160; c++)
        {
```

```
            cout << Y[r][c];
        }
    }
}



int main()
{
    int rHero = 20, cHero = 2;
    int flagJetpack = 0;
    int rarrow1 = 1, carrow1 = 30, dirarrow1 = 1;
    int rarrow2 = 1, carrow2 = 31, dirarrow2 = 1;
    int rarrow3 = 1, carrow3 = 32, dirarrow3 = 1;
    int rlaser1 = 7, claser1 = 18, dilaser1 = 1;
    int relevator = 11, celevator = 150, dielevator = 1, slower = 0;
    int flagHoldRope = 0, cthr = 0, flagMoveRope = 0, flagHeroMoveRope = 0, ctmr = 0;
    int cmvr = 107, prevcr = 0, fix = 1;;
    int renemy3 = 20, cenemy3 = 68, direnemy = 1;
    int flagButton = 0, k = 0, z = 0, flagTouch = 0;
    int flagLadder1up = 0, ctl1up = 0, flagLadder1down = 0, ctl1down = 0, flagLadder2up = 0, ctl2up = 0,
flagLadder2down = 0, ctl2down = 0;
    int flagcoin1 = 0, flagcoin2 = 0, flagcoin3 = 0, flagcoin4 = 0, flagcoin5 = 0, flagcoin6 = 0, flagcoin7 = 0,
flagcoin8 = 0, flagcoin9 = 0, flagcoin10 = 0, ctcoins = 0, check1 = 0, check2 = 0, check3 = 0, check4 = 0,
check5 = 0, check6 = 0, check7 = 0, check8 = 0, check9 = 0, check10 = 0;;
    int flagElevator = 0, ctel = 0, openDoor = 0, prevre = 0, f = 0;
    int flagjump = 0, surfacecheck = 0, surfacecheck1 = 0, ctj = 0, flagfall = 0;
    char ch;
    char y[24][160];

    //enemy 3 platform continuation
    for (int c = 69; c < 80; c++)
    {
        y[19][c] = 220;
    }

    for (;;)
    {
        for (; !_kbhit();)
        {
            Background2(y);
            jetback(y, flagJetpack);
            enemyplatform(y, flagTouch, cenemy3, cHero);
            coinsY(y, flagcoin1, flagcoin2, flagcoin3, flagcoin4, flagcoin5, flagcoin6, flagcoin7, flagcoin8,
flagcoin9, flagcoin10, ctcoins);
            endDoor(y, openDoor);
            if (flagjump == 1)
            {
```

```
            HarakHeroY(y, rHero, cHero, ch, flagJetpack, flagLadder1up, ctl1up, flagLadder1down,
ctl1down, flagLadder2up, ctl2up, flagLadder2down, ctl2down, flagHoldRope, cthr, flagHeroMoveRope,
ctmr, surfacecheck1, flagjump, ctj, surfacecheck, flagfall, flagElevator, dielevator);
        }
        if (flagfall == 1)
        {
            HarakHeroY(y, rHero, cHero, ch, flagJetpack, flagLadder1up, ctl1up, flagLadder1down,
ctl1down, flagLadder2up, ctl2up, flagLadder2down, ctl2down, flagHoldRope, cthr, flagHeroMoveRope,
ctmr, surfacecheck1, flagjump, ctj, surfacecheck, flagfall, flagElevator, dielevator);

        }


        if (flagButton == 0)
        {
            //hole
            for (int c = 69; c < 80; c++)
            {
                y[22][c] = '_';
                y[23][c] = ' ';
            }

            if (z % 10 == 0 && cHero >= 51)
            {
                HarakEnemy3(y, renemy3, cenemy3, direnemy, flagTouch);
            }

            ErsemEnemy3(y, renemy3, cenemy3);

        }
        else
        {
            if (k == 0)
            {
                y[22][70] = '|';
                y[21][70] = '|';
                y[20][70] = '|';
                y[22][78] = '|';
                y[21][78] = '|';
                y[20][78] = '|';
            }
            else
            {
                for (int c = 69; c < 80; c++)
                {
                    y[22][c] = '_';
                    y[23][c] = ' ';
                }
```

```
            }
            renemy3 = 23;
            ErsemEnemy3(y, renemy3, cenemy3);
            k++;
        }
        z++;
        MoveArrow(rarrow1, carrow1, dirarrow1);
        DrawArrows(y, rarrow1, carrow1);
        MoveArrow(rarrow2, carrow2, dirarrow2);
        DrawArrows(y, rarrow2, carrow2);
        MoveArrow(rarrow3, carrow3, dirarrow3);
        DrawArrows(y, rarrow3, carrow3);
        DrawElevator(y, relevator, prevre, flagElevator, openDoor, celevator);
        ErsemHeroY(y, rHero, cHero, flagJetpack, flagHoldRope, flagMoveRope);
        drawlaser(y, rlaser1, claser1);
        movelaser(rlaser1, claser1, dilaser1);


        if (fix == 1 && flagElevator == 1)
        {
            rHero--;
            fix++;
        }

        if (slower % 5 == 0)
        {
            MoveElevator(relevator, celevator, dielevator, rHero, flagElevator);

        }

        slower++;

        DrawRope(y, cmvr, prevcr, flagMoveRope);

        //touch moving arrows
        if (rHero == 4 && cHero == 30 || rHero == 4 && cHero == 31 || rHero == 4 && cHero == 32 ||
rHero == 5 && cHero == 30 || rHero == 5 && cHero == 31 || rHero == 5 && cHero == 32)
        {
            rHero = 20;
            cHero = 3;
            flagJetpack = 0;
        }

        if (flagLadder1up == 1)
        {
            HarakHeroY(y, rHero, cHero, ch, flagJetpack, flagLadder1up, ctl1up, flagLadder1down,
ctl1down, flagLadder2up, ctl2up, flagLadder2down, ctl2down, flagHoldRope, cthr, flagHeroMoveRope,
ctmr, surfacecheck1, flagjump, ctj, surfacecheck, flagfall, flagElevator, dielevator);
```

```
        }

        if (flagLadder1down == 1)
        {
           HarakHeroY(y, rHero, cHero, ch, flagJetpack, flagLadder1up, ctl1up, flagLadder1down,
ctl1down, flagLadder2up, ctl2up, flagLadder2down, ctl2down, flagHoldRope, cthr, flagHeroMoveRope,
ctmr, surfacecheck1, flagjump, ctj, surfacecheck, flagfall, flagElevator, dielevator);
        }

        if (flagLadder2up == 1)
        {
           HarakHeroY(y, rHero, cHero, ch, flagJetpack, flagLadder1up, ctl1up, flagLadder1down,
ctl1down, flagLadder2up, ctl2up, flagLadder2down, ctl2down, flagHoldRope, cthr, flagHeroMoveRope,
ctmr, surfacecheck1, flagjump, ctj, surfacecheck, flagfall, flagElevator, dielevator);
        }

        if (flagLadder2down == 1)
        {
           HarakHeroY(y, rHero, cHero, ch, flagJetpack, flagLadder1up, ctl1up, flagLadder1down,
ctl1down, flagLadder2up, ctl2up, flagLadder2down, ctl2down, flagHoldRope, cthr, flagHeroMoveRope,
ctmr, surfacecheck1, flagjump, ctj, surfacecheck, flagfall, flagElevator, dielevator);
        }

        //coins
        if (rHero == 13 && cHero == 25)
        {
           flagcoin1 = 1;
           if (check1 == 0)
           {
              ctcoins++;
              check1 = 1;
           }
        }

        if (rHero == 10 && cHero == 25)
        {
           flagcoin2 = 1;
           if (check2 == 0)
           {
              ctcoins++;
              check2 = 1;
           }
        }

        if (rHero == 15 && cHero == 38)
        {
           flagcoin3 = 1;
           if (check3 == 0)
```

```
        {
            ctcoins++;
            check3 = 1;
        }
    }

    if (rHero == 19 && cHero == 38)
    {
        flagcoin4 = 1;
        if (check4 == 0)
        {
            ctcoins++;
            check4 = 1;
        }
    }
    if (rHero == 3 && cHero == 59)
    {
        flagcoin5 = 1;
        if (check5 == 0)
        {
            ctcoins++;
            check5 = 1;
        }
    }
    if (rHero == 3 && cHero == 57)
    {
        flagcoin6 = 1;
        if (check6 == 0)
        {
            ctcoins++;
            check6 = 1;
        }
    }

    if (rHero == 3 && cHero == 55)
    {
        flagcoin7 = 1;
        if (check7 == 0)
        {
            ctcoins++;
            check7 = 1;
        }
    }

    if (rHero == 20 && cHero == 139)
    {
        flagcoin8 = 1;
        if (check8 == 0)
```

```
            {
                ctcoins++;
                check8 = 1;
            }
        }

        if (rHero == 20 && cHero == 140)
        {
            flagcoin9 = 1;
            if (check9 == 0)
            {
                ctcoins++;
                check9 = 1;
            }
        }

        if (rHero == 20 && cHero == 141)
        {
            flagcoin10 = 1;
            if (check10 == 0)
            {
                ctcoins++;
                check10 = 1;
            }
        }

        if (flagHoldRope == 1)
        {
            HarakHeroY(y, rHero, cHero, ch, flagJetpack, flagLadder1up, ctl1up, flagLadder1down,
ctl1down, flagLadder2up, ctl2up, flagLadder2down, ctl2down, flagHoldRope, cthr, flagHeroMoveRope,
ctmr, surfacecheck1, flagjump, ctj, surfacecheck, flagfall, flagElevator, dielevator);
        }
        if (flagMoveRope == 1)
        {
            HarakHeroY(y, rHero, cHero, ch, flagJetpack, flagLadder1up, ctl1up, flagLadder1down,
ctl1down, flagLadder2up, ctl2up, flagLadder2down, ctl2down, flagHoldRope, cthr, flagHeroMoveRope,
ctmr, surfacecheck1, flagjump, ctj, surfacecheck, flagfall, flagElevator, dielevator);
            MoveRope(cmvr, prevcr, flagMoveRope, cHero, flagHeroMoveRope, flagMoveRope);
        }

        //return to platform after rope
        if (cHero == 133 && rHero == 9)
        {
            rHero++;
        }

        //raise hand to hold rope
        if (cHero == 103 && rHero == 9)
```

```
        {
           y[rHero][cHero + 1] = '//';
           y[rHero + 1][cHero + 1] = ' ';
        }
        //get the hand back
        if ((cHero == 134 && rHero == 10) || (cHero == 133 && rHero == 10))
        {
           flagHoldRope = 0;
        }

        //the big spike
        if (rHero == 20 && cHero >= 103 && cHero < 135)
        {
           rHero = 20;
           cHero = 2;
        }

        //make the gravity work after using the jetpack
        if (rHero == 20 && cHero == 51)
        {
           flagJetpack = 0;
        }

        //laser
        if (rlaser1 == 21)
        {
           rlaser1 = 7;
        }

       //make the hero die from the laser
        if (rHero == rlaser1 && cHero == 18 || rHero == rlaser1 && cHero == 17 || rHero == rlaser1 &&
cHero == 19)
        {
           rHero = 20;
           cHero = 2;
           flagJetpack = 0;
        }

        if (cHero == 146 && y[rHero + 3][cHero] == -60 || cHero == 147 && y[rHero + 3][cHero] == -60 ||
cHero == 148 && y[rHero + 3][cHero] == -60 || cHero == 149 && y[rHero + 3][cHero] == -60)
        {
           flagElevator = 1;
        }

        cout << "chero:" << cHero << endl << "rhero: " << rHero << endl << "flag elevator: " <<
flagElevator;
        DisplayY(y);
```

```
    }

    ch = _getch();
    HarakHeroY(y, rHero, cHero, ch, flagJetpack, flagLadder1up, ctl1up, flagLadder1down, ctl1down,
flagLadder2up, ctl2up, flagLadder2down, ctl2down, flagHoldRope, cthr, flagHeroMoveRope, ctmr,
surfacecheck1, flagjump, ctj, surfacecheck, flagfall, flagElevator, dielevator);

    if (ch == ' ')
    {
      flagjump = 1;
    }
    if (ch == 'd' && cHero == 12 && rHero == 18)
    {
      flagJetpack = 1;
    }

    if (ch == 'd' && cHero == 51)
    {
      flagJetpack = 2;
      rHero = 20;
    }

    if (ch == 'w' && cHero == 68 && rHero == 8)
    {
      flagLadder1up = 1;
    }

    if (ch == 's' && cHero == 68 && rHero == 3)
    {
      flagLadder1down = 1;

    }

    if (ch == 'w' && cHero == 79 && rHero == 14)
    {
      flagLadder2up = 1;
    }

    if (ch == 's' && cHero == 79 && rHero == 8)
    {
      flagLadder2down = 1;

    }

    if (ch == 'r' && cHero == 102 && rHero == 10)
    {
      flagHoldRope = 1;
    }
```

```
      if (ch == 'd' && cHero == 104 && rHero == 9)
      {
         flagMoveRope = 1;
      }

      if (ch == 'o' && cHero == 153 && rHero == 20)
      {
         openDoor = 1;
      }


      if (cHero == 146 && y[rHero + 3][cHero] == -60 || cHero == 147 && y[rHero + 3][cHero] == -60 ||
cHero == 148 && y[rHero + 3][cHero] == -60 || cHero == 149 && y[rHero + 3][cHero] == -60 || cHero ==
150 && y[rHero + 3][cHero] == -60)
      {
         flagElevator = 1;
      }

      if (ch == 'a' && cHero == 145 || ch == 'a' && cHero == 146)
      {
         flagElevator = 0;
      }

      //open space under enemy
      if (cHero == 62 && rHero == 18)
      {
         flagButton = 1;
      }

      ErsemHeroY(y, rHero, cHero, flagJetpack, flagHoldRope, flagMoveRope);

   }

}
```