rr

**МIНIСТЕРСТВО ОСВIТИ І НАУКИ УКРАЇНИ**

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ**

**“КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ**

**ІМЕНІ ІГОРЯ СІКОРСЬКОГО”**

**Факультет прикладної математики**

**Кафедра програмного забезпечення комп’ютерних систем**

**Лабораторна робота № 6**

**з дисципліни “ Основи програмування ”**

**тема “СТРУКТУРИ ДАНИХ”**

|  |  |  |
| --- | --- | --- |
| **Виконав**  **студент I курсу**  **групи КП-83**  **Мортіков Владислав Євгенович**  **варіант №8** |  | **Перевірив**  **“\_\_\_\_” “\_\_\_\_\_\_\_\_\_\_\_\_” 20\_\_\_ р.**  **викладач**  **Гадиняк Руслан Анатолійович**  **(прізвище, ім’я, по батькові)** |

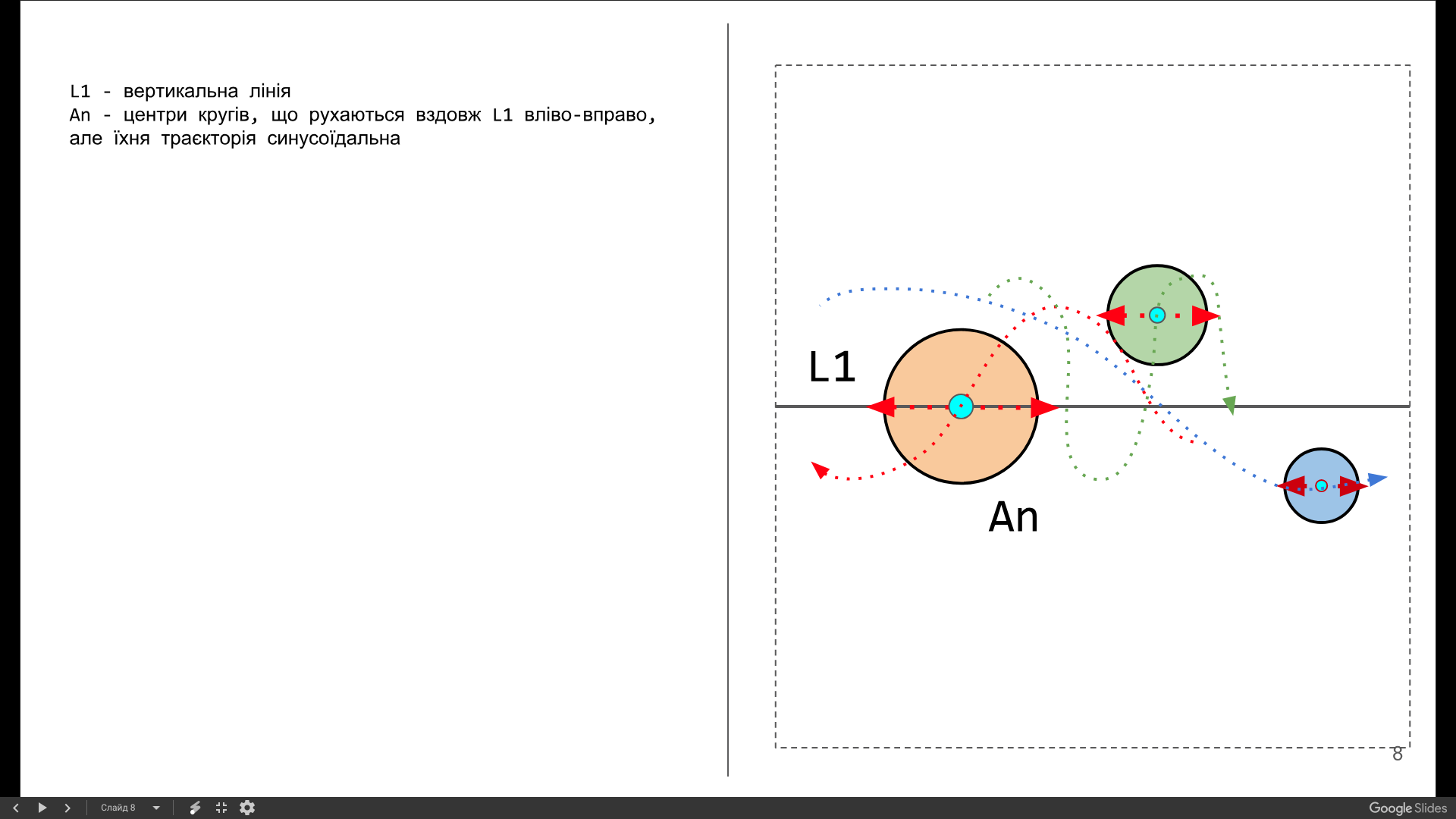
**Київ 2018**

**Мета роботи**

Навчитися працювати зі структурами даних.

**Постановка завдання**

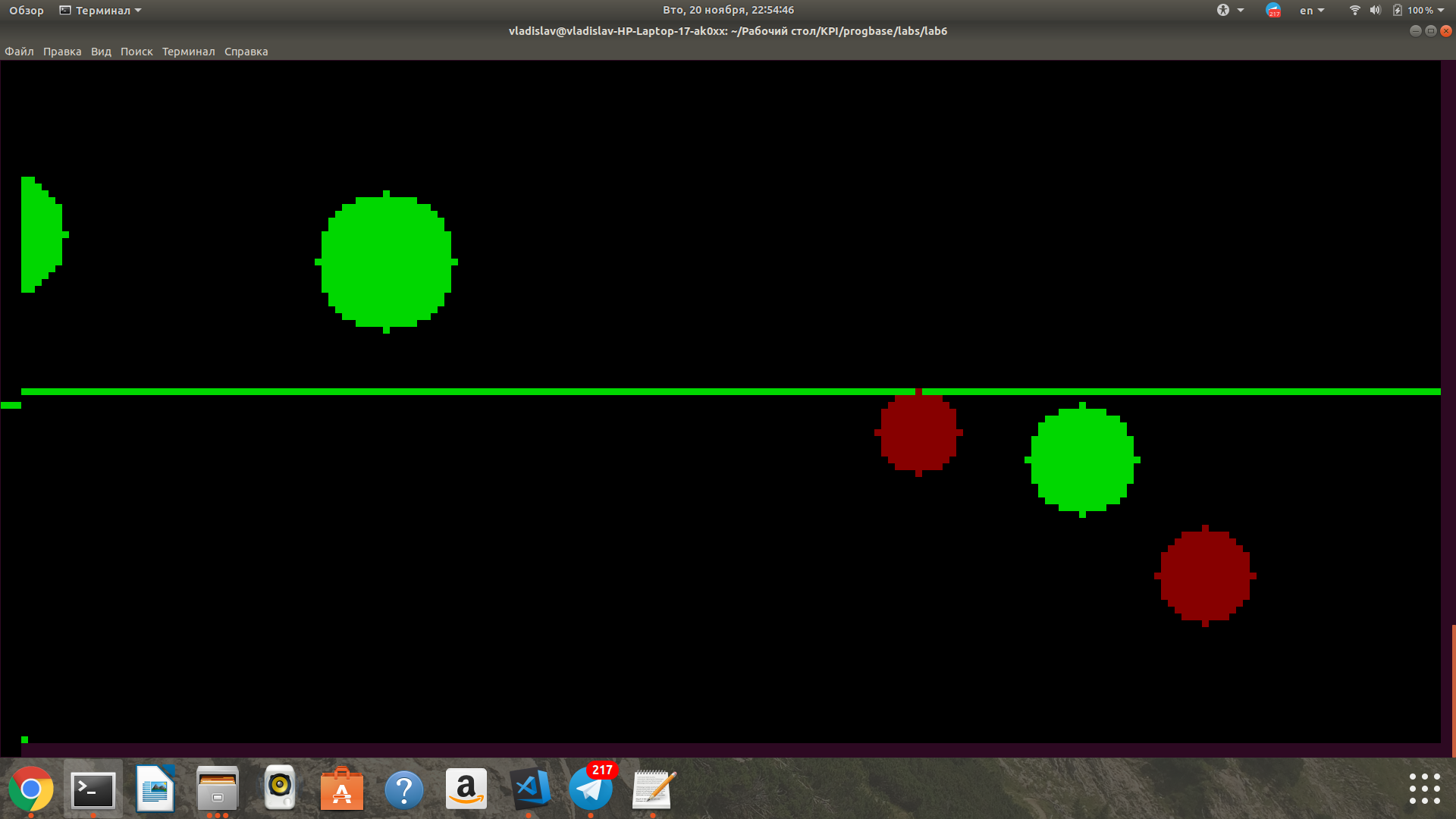
За допомогою функцій модуля **Canvas** бібліотеки **libprogbase** (progbase/canvas.h, версія бібліотеки не менше [0.6.0](https://github.com/PublicHadyniak/libprogbase/releases/tag/0.6.0)) намалювати у терміналі графічну сцену, що складається із графічних об'єктів та піддається певним правилам взаємодії за варіантом.

Виділити відповідні типи даних графічних об'єктів як структури і використовувати змінні цих структур для зберігання їх параметрів. Однотипні змінні об'єднати у масиви структур.

**Тексти коду програм**

|  |
| --- |
| **main.c** |
| #include <stdio.h>  #include <math.h>  #include <stdlib.h>  #include <time.h>  #include <stdbool.h>  #include <progbase.h>  #include <progbase/console.h>  #include <progbase/canvas.h>  struct vec2D  {  float x;  float y;  };  struct ball  {  struct vec2D loc;  int rad;  struct vec2D vel;  char colour1;  char colour2;  char colour3;  float k;  };  int main()  {  srand(time(0));  Console\_clear();  const float pi = 3.14;  int N = 5;  struct ConsoleSize consoleSize = Console\_size();  const int widthPixels = consoleSize.columns;  const int heightPixels = consoleSize.rows \* 2;  struct ball shar[N];  int FPS = 30;  for (int i = 0; i < N; i++)  {  shar[i].loc.x = rand() % ((widthPixels / 2 + 20) - 20 + 1) + 20;  shar[i].loc.y = rand() % ((heightPixels / 2 + 20) - 20 + 1) + 20;  shar[i].rad = rand() % (10 - 5 + 1) + 5;  shar[i].colour1 = rand() % 254 + 1;  shar[i].colour2 = rand() % 254 + 1;  shar[i].colour3 = rand() % 254 + 1;  shar[i].k = 10;  shar[i].vel.x = pi \* (rand() % (10 - 1 / 10 + 1) + 1 / 10);  shar[i].vel.y = shar[i].k \* sin(shar[i].vel.x);  }  Canvas\_setSize(widthPixels, heightPixels);  Canvas\_invertYOrientation();  Canvas\_setOrigin(2, 4);  bool islab6Running = true;  int x1 = 0;  char workingLab = Console\_getChar();  printf("You entered: '%c'\n", workingLab);  while (islab6Running)  {  switch (workingLab)  {  case '0':  {  islab6Running = false;  }  break;  default:  {  Canvas\_beginDraw();  Canvas\_setColorRGB(0, 250, 0);  Canvas\_strokeLine(0, (heightPixels / 2), widthPixels, (heightPixels / 2));  Canvas\_putPixel(0, 0);  for (int i = 0; i < N; i++)  {  if (i <= 0.5 \* N)  {  if (shar[i].loc.x + shar[i].rad >= widthPixels || shar[i].loc.x - shar[i].rad <= 0)  {  shar[i].vel.x = -shar[i].vel.x;  }  if (shar[i].loc.y + shar[i].rad >= heightPixels || shar[i].loc.y - shar[i].rad <= 0)  {  shar[i].vel.y = 0.5 \* shar[i].vel.y;  }  x1++;  shar[i].loc.x += shar[i].vel.x;  shar[i].loc.y += 10 \* sin(x1);  Canvas\_setColorRGB(shar[i].colour1, shar[i].colour2, shar[i].colour3);  Canvas\_fillCircle(shar[i].loc.x, shar[i].loc.y, shar[i].rad);  }  else  {  shar[i].k = -shar[i].k;  if (shar[i].loc.x + shar[i].rad >= widthPixels || shar[i].loc.x - shar[i].rad <= 0)  {  shar[i].vel.x = -shar[i].vel.x;  }  if (shar[i].loc.y + shar[i].rad >= heightPixels || shar[i].loc.y - shar[i].rad <= 0)  {  shar[i].vel.y = 0.5 \* shar[i].vel.y;  }  x1++;  shar[i].loc.x -= shar[i].vel.x;  shar[i].loc.y += 10 \* sin(x1);  Canvas\_setColorRGB(shar[i].colour1, shar[i].colour2, shar[i].colour3);  Canvas\_fillCircle(shar[i].loc.x, shar[i].loc.y, shar[i].rad);  }  }  sleepMillis(FPS);  Canvas\_endDraw();  }  // if (islab6Running == false)  // {  // break;  // }  }  }  return 0;  } |

**￼Приклади результатів**

****

**Висновки**

Виконавши дану лабораторну роботу ми навчилися працювати із структурами даних. Застосували на практиці різні види циклічних конструкцій (з передумовою, із лічильником) при роботі із бібліотекою canvas.

Компіляція всього коду відбувалася за допомогою компілятора gcc.