

## Лабораторна робота Bash

### РОЗРОБКА ГРИ НА MOBI BASH

**Мета:** використовуючи команди bash створити гру.

#### Хід роботи:

Обрано для створення гру змійка. Правила гри наступні: у гравця є поле, в межах якого він може рухати змійку. Коли гравець входить в край поля з одної сторони, голова змійки переноситься на протилежну сторону. На полі з'являються яблука, які змійка має «їсти» аби збільшуватися. Якщо голова змійки стикається з частиною її тіла – гра закінчується поразкою. Мета гри – отримання якнайвищого рахунку, який зберігається між ігровими сесіями.

Оскільки гра буде консольною, то програмування почато зі створення функцій взаємодії з консоллю, в тому числі друк символу в певному місці та завершення виконання програми після натиснення будь-якої кнопки.

Лістинг файлу console.sh:

```
#!/bin/bash

# Перенесення курсора на задану позицію
move_cursor_to_position() {
    local row="$1"
    local col="$2"

    # Move cursor to the specified row and column
    echo -e -n "\e[${row}];${col}H"
}

# Запис символу на задану позицію
write_char_at_position() {
    local row="$1"
    local col="$2"
    local char="$3"

    move_cursor_to_position "$row" "$col"

    echo -n "$char"
}
```

					ДУ «Житомирська політехніка».23.121.06.000 – ЛрBash			
Змн.	Арк.	№ докум.	Підпис	Дата				
Розроб.		Каліновський Д.			Звіт з лабораторної роботи		Лім.	Арк.
Перевір.		Єфремов Ю.М.						1
Керівник							Аркушів	
Н. контр.							8	
Зав. каф.							ФІКТ Гр. ІПЗк-21-1	

```

# Відображення границь вікна
draw_window_borders() {
    local height="$1"
    local width="$2"
    local char="$3"

    # Малювання вертикальних границь
    for ((i=0; i < height; i++)); do
        write_char_at_position "$i" 0 "$char"
        write_char_at_position "$i" "$((width - 1))" "$char"
    done

    # Малювання горизонтальних границь
    for ((i=0; i < width; i++)); do
        write_char_at_position 0 "$i" "$char"
        write_char_at_position "$((height - 1))" "$i" "$char"
    done
}

# Завершення гри
exit_on_input() {
    local row="$1"

    move_cursor_to_position "$row" 0
    # Відновлення видимості курсора
    tput cnorm
    # Відображення повідомлення про завершення гри
    read -n 1 -s -r -p "Press any key to exit..."

    clear
    move_cursor_to_position 0 0
}

```

Наступним кроком створено файл `variables.sh` для локалізації усіх змінних, значення яких можуть бути змінені з часом, як розміри поля, символи поля, змійки та яблука і таке інше.

Через особливості `bash` та відсутність вбудованих структур даних, обрано реалізацію збереження координат у рядку. Для операцій додання, пошуку та видалення координат створено ряд функцій у файлі `coordinates_functions.sh`.

Лістинг файлу `coordinates_functions.sh`:

```

#!/bin/bash
# Обмежена реалізація структури Черга (FIFO) на основі рядка, оскільки можливості мови bash обмежені

# Додавання координати в кінець списку координат

```

		Каліновський Д.			ДУ «Житомирська політехніка».23.121.06.000 – ЛрBash	Арк.
		Єфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		2

```

coords_append() {
    local x=$1
    local y=$2
    coordinates+="($x,$y) "
}

# Перевірка чи координати містяться в списку координат
coords_contain() {
    local x="$1"
    local y="$2"
    local element_to_check="($x,$y)"
    # echo "Ele: $element_to_check"
    # Перевірка чи координати містяться в списку координат за допомогою регулярного виразу
    if [[ $coordinates == *"$element_to_check"* ]]; then
        return 0
    else
        return 1
    fi
}

# Видалення найстарішої координати зі списку координат
coords_pop() {
    # Отримання найстарішої координати
    local temp="${coordinates%% *}"

    # Видалення дужок з координати та занесення в глобальну змінну
    temp="${temp//(/}"
    last_coords="${temp//)/}"

    # Видалення найстарішої координати зі списку координат
    coordinates="${coordinates#* }"
}

```

Створено файл game.sh з циклом гри, що опрацьовує ввід користувача, виведення інформації, створення яблук, процесу їх «поїдання», переміщення змійки та завершення гри.

Лістинг файлу game.sh:

```

#!/bin/bash

set -o allexport
source console.sh
source variables.sh
source coordinates_functions.sh
set +o allexport

# Оголошення глобальних змінних
export coordinates="" last_coords=""

```

		Каліновський Д.			ДУ «Житомирська політехніка».23.121.06.000 – ЛрBash	Арк.
		Єфремов Ю.М.				3
Змн.	Арк.	№ докум.	Підпис	Дата		

```

apple_x=0
apple_y=0
score=0
highscore=0

# Отримання рахунку з файлу
if [ -f "$HIGHSCORE_FILE" ] ; then
    highscore=$(cat "$HIGHSCORE_FILE")
fi

# Створення яблука
spawn_apple() {
    apple_x=$((RANDOM % (WINDOW_HEIGHT-3) + 2))
    apple_y=$((RANDOM % (WINDOW_WIDTH-3) + 2))

    # Якщо координати яблука збігаються з координатами гравця, то генеруємо нові координати
    while coords_contain "$apple_x" "$apple_y" ; do
        apple_x=$((RANDOM % (WINDOW_HEIGHT-3) + 2))
        apple_y=$((RANDOM % (WINDOW_WIDTH-3) + 2))
    done

    # Відображення яблука
    write_char_at_position "$apple_x" "$apple_y" "$APPLE_CHAR"
}

# Головний цикл гри
run_game_loop() {
    # Встановлення початкових координат гравця в центрі вікна
    local x=$((WINDOW_HEIGHT/2))
    local y=$((WINDOW_WIDTH/2))

    # Створення гравця
    coords_append "$x" "$y"
    write_char_at_position "$x" "$y" "$PLAYER_CHAR"

    # Створення яблука
    spawn_apple

    # Відображення рекорду
    write_char_at_position $((WINDOW_HEIGHT/2 + 1)) $((WINDOW_WIDTH+5)) "Highest score:
$highscore"

    # Цикл гри
    while true; do
        # Зчитування вводу з клавіатури з таймаутом
        read -r -t "$GAME_TICK_LENGTH" -n 1 -s holder && input="$holder"

        # Якщо ввід не є прийнятним, він ігнорується

```

		Каліновський Д.			ДУ «Житомирська політехніка».23.121.06.000 – ЛрBash	Арк.
		Єфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		4

```

if [[ ! $input =~ [^wasde] ]]; then
    direction="$input"
fi

# В залежності від напрямку руху гравця, змінюються його координати.
# Також тут реалізовано вихід з гри та ігровування вводу доки він не буде прийнятим
case "$direction" in
    w) ((x--));;
    s) ((x++));;
    a) ((y--));;
    d) ((y++));;
    e) break;;
    *) continue
       # *)
esac

# Якщо відлагоджувальний режим включений, то відображається додаткова інформація
if [ "$DEBUG" -eq 0 ] ; then
    write_char_at_position $((WINDOW_HEIGHT+1)) "$WINDOW_WIDTH" "LC: $last_coords"
    write_char_at_position $((WINDOW_HEIGHT+2)) "$WINDOW_WIDTH" "RX: $rx"
    write_char_at_position $((WINDOW_HEIGHT+3)) "$WINDOW_WIDTH" "RY: $ry"
    write_char_at_position $((WINDOW_HEIGHT+4)) "$WINDOW_WIDTH" "Coords: $coordinates"
fi

# Відображення рахунку
write_char_at_position $((WINDOW_HEIGHT/2)) $((WINDOW_WIDTH+5)) "Score: $score"

# Якщо гравець не з'їв яблуко, то видаляється хвіст змії
if [ "$x" -ne "$apple_x" ] || [ "$y" -ne "$apple_y" ] ; then
    coords_pop
    IFS=',' read -r rx ry <<< "$last_coords"
    write_char_at_position "$rx" "$ry" " "
fi

# Перевірка чи гравець не вийшов за межі вікна та перенесення на протилежний бік вікна
if [ "$x" -lt 2 ]; then
    x=$((WINDOW_HEIGHT-2))
fi

if [ "$x" -gt $((WINDOW_HEIGHT-2)) ]; then
    x=2
fi

if [ "$y" -lt 2 ]; then
    y=$((WINDOW_WIDTH-2))
fi

if [ "$y" -gt $((WINDOW_WIDTH-2)) ]; then
    y=2
fi

```

		Каліновський Д.			ДУ «Житомирська політехніка».23.121.06.000 – ЛрBash	Арк.
		Єфремов Ю.М.				5
Змн.	Арк.	№ докум.	Підпис	Дата		

```

# Перевірка чи гравець не зіткнувся з собою (тобто програв)
if coords_contain "$x" "$y" ; then
    write_char_at_position $((WINDOW_HEIGHT/2)) $((WINDOW_WIDTH/2-3)) "*DEAD*"
    break
fi

# Додавання нової координати гравця та відображення нової голови змії
coords_append "$x" "$y"
write_char_at_position "$x" "$y" "$PLAYER_CHAR"

# Якщо гравець з'їв яблуко, то генерується нове яблуко та збільшується рахунок
if [ "$x" -eq "$apple_x" ] && [ "$y" -eq "$apple_y" ] ; then
    spawn_apple
    score=$((score+1))
fi

# Видалення хвоста змії
if ! coords_contain "$rx" "$ry" ; then
    write_char_at_position "$rx" "$ry" " "
fi
done

if [ "$highscore" -lt "$score" ] ; then
    echo "$score" > "$HIGHSCORE_FILE"
fi
}

```

Створено файл main.sh, який користувач буде запускати для початку гри. В ньому курсор в консолі встановлюється прозорим, консоль очищується та друкується ігрове поле, після чого запускається ігровий цикл. Коли ігровий цикл закінчується, виконання файлу завершується викликом функції, що повертає видимість курсору та повідомляє користувача, що виконання програми завершиться після того, як він натисне будь-яку кнопку, після чого консоль очищується, а курсор переноситься на початок консолі.

Лістинг файлу main.sh:

```

#!/bin/bash

# Імпортування функцій та змінних з інших файлів
set -o allexport
source console.sh
source variables.sh
source game.sh
set +o allexport

# Встановлення прозорого курсора

```

		Каліновський Д.			ДУ «Житомирська політехніка».23.121.06.000 – ЛpBash	Арк.
		Єфремов Ю.М.				6
Змн.	Арк.	№ докум.	Підпис	Дата		

```

tput civis

clear

# Встановлення ігрового розмірів вікна
draw_window_borders "$WINDOW_HEIGHT" "$WINDOW_WIDTH" "$WINDOW_CHAR"

# Запуск головного циклу гри
run_game_loop

# Завершення гри
exit_on_input "$WINDOW_HEIGHT"

```

Після написання застосунків протестовано. Рисунок процесу гри:

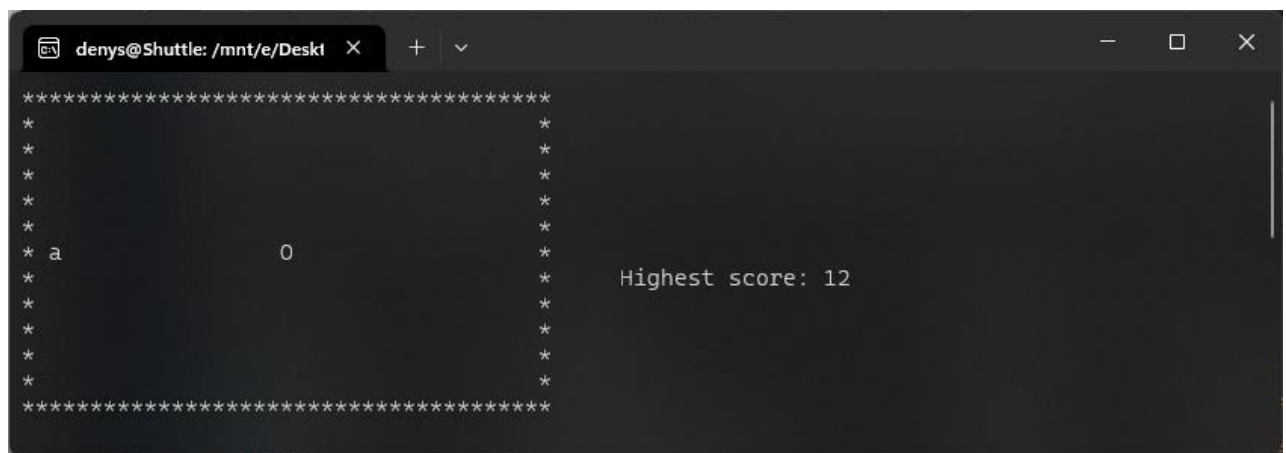


Рисунок 1. Початковий екран гри.

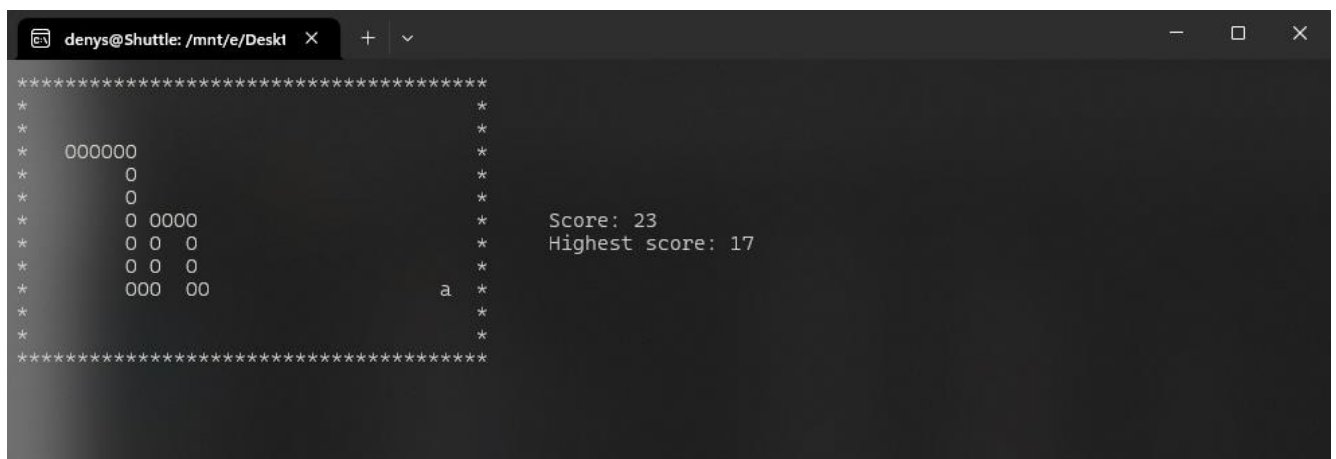


Рисунок 2. Процес гри.

		Каліновський Д.			ДУ «Житомирська політехніка».23.121.06.000 – ЛpBash	Арк.
		Єфремов Ю.М.				7
Змн.	Арк.	№ докум.	Підпис	Дата		

```

denys@Shuttle: /mnt/e/Desk1
*****
*                                     *
*                                     *
*          000000                    *
*          0 0                      *
*          0000                     *
*      *DEAD*                        *   Score: 12
*          a                         *   Highest score: 23
*                                     *
*                                     *
*                                     *
*                                     *
*                                     *
*****
Press any key to exit...|

```

Рисунок 3. Програш.

Файли програми викладено на Github за посиланням:

<https://github.com/DeNice-r/bash-snake>

**Висновок:** під час виконання даної лабораторної роботи я вивчив більшість синтаксичних особливостей bash, а також на початковому рівні навчився створювати програми, що включають в себе маніпуляції з даними різних типів та файлами.

		Каліновський Д.			ДУ «Житомирська політехніка».23.121.06.000 – ЛрBash	Арк.
		Єфремов Ю.М.				8
Змн.	Арк.	№ докум.	Підпис	Дата		