메타버스 프로그래밍 1기

2개월차 레벨 테스트

이름 : 김동현

해당 테스트는 객관식, 단답형, 서술식, 코드 작성의 형태로 구성되어 있으며, 코드 작성을 제외한 문제들은 문제지에 해답을 작성한다.

코드 작성 형식은 Github Repository를 생성 후 아래의 형식에 맞춰 답안을 올린다.

* Repository 이름 : Metaverse\_01\_LevelTest
* Github Link는 담임교수에게 바로 전달
* 문제마다 파일을 따로 작성, 파일 이름 : Solution\_01.cpp, Solution\_02.cpp …
* 파일 상단에 주석으로 [문제를 파악한 내용], [해결 아이디어] 등을 작성

1. 컴퓨터를 이용하여 데이터를 처리할 수 있다. 다음 중 컴퓨터가 처리할 수 있는 데이터의 형태는 무엇인가? [ B ]
   1. 문자 B. 숫자 C. 전파 D. 소리
2. 십진수 15를 이진수 4자리(4bit)로 표현해 보자.

[1111]

1. 위의 4bit 숫자를 X라고 했을 때, 아래의 연산자를 통한 결과값을 십진수로 각각 적어 보자.
   1. X << 1 [ 30 ]
   2. X << 2 [ 60 ]
   3. X << 3 [ 120 ]
2. 다음의 프로그램 개발 과정을 순서대로 정렬해 보자.

[ F -> E -> A -> D-> C-> B ]

* 1. 코딩
  2. 유지보수
  3. 실행과 디버깅
  4. 컴파일과 링크
  5. 알고리즘의 개발
  6. 요구사항분석

1. 프로그래밍을 하다 보면 컴파일 오류, 런타임 오류, 논리 오류를 발생시키는 코드를 작성하게 될 때가 있다. 각 오류에 대해 서술하고 해당 오류가 발생할 예제 코드를 적어보자.
   1. 컴파일 오류 :

#define CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

int main(void)

{

const int a = 10;

a = 20;

return 0;

}

const 때문에 변수 a를 바꿀 수 없습니다. 문법 오류입니다.

* 1. 런타임 오류 :

#define CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

int main(void)

{

int a = 10;

int b = 2;

int c;

c = a / b;

while (1)

{

printf("%d", c);

}

return 0;

}

무한루프일 때 런타임오류라 할 수 있습니다.

* 1. 논리 오류 :

#define CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

int main(void)

{

int a = 2;

int b = 3;

int c = 10;

int d;

// a와 b의 합에 c를 곱하고싶었다

d = a + b \* c;

printf("%d", d);

return 0;

}

정상적으로 동작은 하지만 원하는 값을 얻지 못하는 것

1. 다음의 코드에서 에러가 발생하는 이유는 무엇인가?

int main(void)

{

int\* aP;

\*aP = 10;

    return 0;

}

}

변수 aP가 초기화 되지 않았습니다.

1. 다음의 코드에서 에러가 발생하는 부분과 그 이유는 무엇인가?

int main(void)

{

    const int a = 10;

    a = 20;

    int const b = 20;

    return 0;

}

}

a = 20; 부분이 잘못되었습니다.

const로 선언되있는 a변수의 값을 바꾸려 했기 때문입니다

1. 다음의 코드를 보고 A,B 중 어느 부분이 실행될지 맞추고 이유를 서술하시오.

int\* p = NULL;

if (p)

{

    // A

}

else

{

    // B

}

B가 실행됩니다.

Int\* p가 비어있기 때문입니다.

1. 다음의 코드를 보고 빈 칸에 들어갈 적절한 배열포인터를 선언해 보자.

[ int\* parr = arr; ]

[ ]; // pointer to array of int

int arr[5];

parr = &arr; // 가능

int arr2[4][5];

parr = arr2; // 가능

1. 다음의 코드에서 n 의 값은 무엇인가?

[ 3 ]

int arr[5] = { 1, 2, 3, 4, 5 };

int\* p = arr

p++;

int n = \*++p

1. 다음의 코드를 보고 구조체의 BestFriend 멤버에 접근하는 코드를 두가지 방식으로 구현해보자.

#define CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

struct Student

{

int Age;

enum { A, B, O, AB } BloodType;

char Name[24];

struct Student\* BestFriend;

};

int main(void)

{

struct Student seonmun;

struct Student\* p = &seonmun;

seonmun.BestFriend; // 1번째

p->BestFriend; // 2번째

}

struct Student

{

    int Age;

    enum { A, B, O, AB } BloodType;

    char Name[24];

    struct Student\* BestFriend;

};

Student seonmun = { 20, A, "최선문", NULL };

[ 코드 구현 문제 ]

1. 포인터와 레퍼런스를 활용해 함수 외부에서 선언된 두 정수형 변수의 값을 서로 바꾸는 Swap 함수를 각각 구현하시오.
   1. void swap(int\* a, int\* b);

#define CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

void swap(int\* a, int\* b)

{

int \_a, \_b;

int \_c;

\_a = a;

\_b = b;

\_c = \_a;

\_a = \_b;

\_b = \_c;

printf("%d %d", \_a, \_b);

}

* 1. void swap(int& a, int& b);

1. 1000미만의 3의 승수를 차례대로 출력하는 코드를 구현 하시오.

#define CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

// 변수 a를 만들고 1로 초기화를 해준 후 for 문을 사용하여

int main(void)

{

int a = 1;

for (int i = 1; a < 1000; i \* 3)

{

a \*= 3;

if (a >= 1000)

{

break;

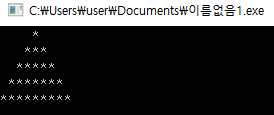
}

printf("%d\n", a);

}

}

1. 다음의 형태로 \* 을 출력하는 코드를 구현 하시오.



#define CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

int main(void)

{

for (int i = 0; i < 5; ++i)

{

for (int j = 5; i < j - 1; --j)

{

printf(" ");

}

for (int j = 0; j < i \* 2 + 1; ++j)

{

printf("\*");

}

printf("\n");

}

}

1. 배열의 값을 오름차순으로 정렬해서 반환하는 함수를 구현 하시오.

int arr[5] = { 5, 2, 4, 1, 3 };

void sortArray(int\* arr, int n)

{

…

}

sortArray(arr, 5);

// arr : [ 1, 2, 3, 4, 5 ]

#define CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <stdlib.h>

void sortArray(int\* arr, int n)

{

int crr[1024] = { 0 };

int min = n;

// 배열 복사

for (int i = 0; i < n; i++)

{

crr[i] = arr[i];

}

for (int i = 0; arr[i] != '/n'; ++i)

{

if (min > arr[i])

{

min = arr[i];

}

}

for (int i = 0; i < n; ++i)

{

printf("%d ", crr[i]);

}

}

int main(void)

{

int arr[5] = { 5, 2, 4, 1, 3 };

sortArray(arr, 5);

}

모르겠습니다. 알거같은데 햇갈립니다.

1. 로또 번호 생성기를 구현 하시오.
   1. 최대 번호값과 생성 숫자 개수를 입력하면 해당 조건 내에서 중복되지 않는 숫자를 출력하는 로또 번호 생성기를 구현.
   2. 최대값 : 60, 생성숫자 : 10 을 입력하면 1 ~ 60 사이의 중복되지 않는 숫자 10개를 생성한다.

#define CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

int main(void)

{

srand(time(NULL));

int max\_num = 0; // 최대 번호값;

int creat\_num = 0; // 생성숫자 개수;

int count = 0; // 돌린 횟수

int num = 0; // 출력 숫자;

int arr[1000] = { 0 };

scanf\_s("%d %d", &max\_num, &creat\_num);

for (count; count < creat\_num; ++count)

{

num = rand() % max\_num + 1;

arr[count] = num;

for (int i = 0; i < count; ++i)

{

if (arr[i] == arr[count])

{

arr[count] = rand() % max\_num + 1;

i = -1;

}

}

printf("%d ", arr[count]);

}

}

1. Joker를 포함한 53장의 카드를 섞은 후, 두 명의 플레이어에게 7장씩의 카드를 배분하는 프로그램을 만드시오.
   1. ♠♣♥◆ 네 개의 기호 + 기호 별 각각 A, 2 ~ 10, J, Q, K 값의 카드 존재
   2. 출력예시) Player 1 : ♥3, Joker, ♣A, ♣J, ◆1, ◆10, ♠4  
       Player 2 : ♥6, ◆2, ◆5, ♠Q, ♠K, ♣2, ♣3

#define CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <stdlib.h>

#include <stdbool.h>

#include <time.h>

int main(void)

{

char\* card\_game[53] = {

"♥A", "♥2", "♥3", "♥4", "♥5", "♥6", "♥7", "♥8", "♥9", "♥10", "♥J", "♥Q", "♥K",

"♣A", "♣2", "♣3", "♣4", "♣5", "♣6", "♣7", "♣8", "♣9", "♣10", "♣J", "♣Q", "♣K",

"♠A", "♠2", "♠3", "♠4", "♠5", "♠6", "♠7", "♠8", "♠9", "♠10", "♠J", "♠Q", "♠K",

"◆A", "◆2", "◆3", "◆4", "◆5", "◆6", "◆7", "◆8", "◆9", "◆10", "◆J", "◆Q", "◆K", "Joker"

};

char\* player1\_card[7] = { 0 };

char\* player2\_card[7] = { 0 };

int symbol1 = 0;

int symbol2 = 0;

int symbol3 = 0;

srand(time(NULL));

for (int turn = 0; turn < 7; ++turn)

{

symbol1 = rand() % 53;

symbol2 = rand() % 53;

player1\_card[turn] = card\_game[symbol1];

player2\_card[turn] = card\_game[symbol2];

for (int i = 0; i < turn; ++i)

{

if (player1\_card[i] == player1\_card[turn])

{

symbol1 = rand() % 53;

player1\_card[turn] = card\_game[symbol1];

i = -1;

}

}

for (int i = 0; i < turn; ++i)

{

if (player2\_card[i] == player2\_card[turn] || player1\_card[i] == player2\_card[turn])

{

symbol2 = rand() % 53;

player2\_card[turn] = card\_game[symbol2];

i = -1;

}

}

}

for (int i = 0; i < 7; ++i)

{

for (int j = 0; j < 7; ++j)

{

if (player1\_card[i] == player2\_card[j])

{

symbol3 = rand() % 53;

player2\_card[j] = card\_game[symbol3];

}

}

}

printf("Player 1 : ");

for (int i = 0; i < 7; ++i)

{

printf("%s ", player1\_card[i]);

}

printf("\n");

printf("Player 2 : ");

for (int i = 0; i < 7; ++i)

{

printf("%s ", player2\_card[i]);

}

}

1. 1 ~ 25까지의 숫자가 랜덤하게 배치되는 5X5의 빙고게임 판을 생성한 후, 숫자 입력을 받을 때마다 해당 숫자를 지워주고 가로, 세로, 대각선의 빙고 줄이 완성될 때마다 빙고 개수를 갱신해서 표시해 주는 빙고 게임을 제작 하시오.

출력예시)

19 79 16 23 20

5 13 10 14 24

6 12 18 2 4

21 3 7 11 1

15 25 22 8 9

현재 0줄의 빙고가 완성되었습니다.

숫자를 입력해 주세요 :

#define CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <stdlib.h>

#include <stdbool.h>

#include <time.h>

int main(void)

{

srand(time(NULL));

int bord[5][5] = { 0 };

int num = 0;

int change\_num = 0;

int arr[5][5] = {0};

for (int i = 0; i < 5; ++i)

{

for (int j = 0; j < 5; ++j)

{

num = rand() % 25 + 1;

bord[i][j] = num;

arr[i][j] = num;

for (int a = 0; a < 5; ++a)

{

for (int b = 0; b < 5; ++b)

{

if (bord[i][j] == arr[a][b])

{

change\_num = rand() % 25 + 1;

arr[a][b] = change\_num;

}

}

}

printf("%d\t", bord[i][j]);

}

printf("\n");

}

}

1. N \* N 달팽이 배열 알고리즘을 구현 하시오.
   1. 이차원 배열의 크기를 결정할 N을 정수로 입력 받는다.
   2. 출력예시)

배열의 크기를 입력하세요 : \_5

1 2 3 4 5

16 17 18 19 6

15 24 25 20 7

14 23 22 21 8

13 12 11 10 9