메타버스 프로그래밍 1기

2개월차 레벨 테스트

이름 : 안중재

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 : https://github.com/bbno1105/Metaverse\_01\_LevelTest
* 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 : 컴파일 단계에서 작성한 코드가 문법적인 문제로 발생하는 오류

|  |
| --- |
| int number;  cout << number << endl; // 초기화되지 않은 변수의 사용 |

* 1. 런타임 오류 : 실행 단계에서 불가능한 접근 또는 계산이 이루어져 발생한 오류

|  |
| --- |
| int number[2] = { 0 , 100 };  std::cout << number[1000] << endl; // 액세스 위반 발생  std::cout << number[1] / number[0] << endl; // 0으로 나눔. |

* 1. 논리 오류 : 문법, 실행에는 문제가 없으나 의도하지 않은 작동으로 발생한 오류

|  |
| --- |
| if (false)  {  cout << "Hello World!" << endl; // 실행되지 않는 코드  } |

1. 다음의 코드에서 에러가 발생하는 이유는 무엇인가?

int main(void)

{

int\* aP;

\*aP = 10;

    return 0;

}

}

포인터 변수 aP가 가리킬 int 타입의 대상이 정의되지 않았기에 오류가 발생하였습니다.

1. 다음의 코드에서 에러가 발생하는 부분과 그 이유는 무엇인가?

int main(void)

{

    const int a = 10;

    a = 20;

    int const b = 20;

    return 0;

}

}

const로 선언된 객체는 값이 정의된 이후 값의 변경이 제한된다.

따라서 const int로 선언된 객체 a의 값은 10으로 정의된 이후 20이라는 값을 할당하는 것이 불가능하다.

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

int\* p = NULL;

if (p)

{

    // A

}

else

{

    // B

}

포인터 타입의 p는 NULL 값으로 정의되어 (x64 기준) 0000000000000000 의 주소값을 갖게 된다. 조건식에서 0은 false를 나타내기 때문에 조건식은 else로 접근하여 B가 실행된다.

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

[ int\* parr ]

[ ]; // 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.

[ seonmun.BestFriend-> ]

[ \*(seonmun.BestFriend). ]

struct Student

{

    int Age;

    enum { A, B, O, AB } BloodType;

    char Name[24];

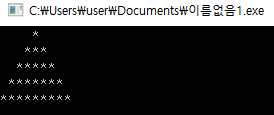
    struct Student\* BestFriend;

};

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);
   2. void swap(int& a, int& b);
2. 1000미만의 3의 승수를 차례대로 출력하는 코드를 구현 하시오.
3. 다음의 형태로 \* 을 출력하는 코드를 구현 하시오.



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 ]

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.
2. 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
3. 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 :

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