**고급시스템프로그래밍 2주차 실시간수업 과제**

|  |
| --- |
| 제출일: 2021.03.18  이름: 김현용 |

1. 다음 프로그램을 수정하여(“// 수정” 부분만 수정) 모든 단계를 통과하도록 하시오.

|  |
| --- |
| #include <stdio.h>  #include <stdlib.h>  #include <stdint.h>  int main() {  int x,y;  unsigned ux,uy;  x = 0; // 수정  if (x<0 && x\*2>=0)  printf("#1 Passed! (x:%d)\n", x);  else  printf("#1 Failed! (x:%d)\n", x);  ux = 0; // 수정  if (ux > -2)  printf("#1 Passed! (ux:%u)\n", ux);  else  printf("#1 Failed! (ux:%u)\n", ux);  x = 0; // 수정  y = 0; // 수정  if (x>y && -x>-y)  printf("#1 Passed! (x,y:%d,%d)\n", x,y);  else  printf("#1 Failed! (x,y:%d,%d)\n", x,y);  x = 0; // 수정  y = 0; // 수정  if (x>>3 == x/8 && y>>3 != y/8)  printf("#1 Passed! (x,y:%d,%d)\n", x,y);  else  printf("#1 Failed! (x,y:%d,%d)\n", x,y);  } |

A1\_1) 소스 코드

#include <stdio.h>

#include <stdlib.h>

#include <stdint.h>

#define TMAX 2147483647

#define UMAX 4294967295

#define TMIN -2147483648

#define UMIN 0

int main() {

int x,y;

unsigned ux,uy;

x = TMIN; // 수정

if (x<0 && x\*2>=0)

printf("#1 Passed! (x:%d)\n", x);

else

printf("#1 Failed! (x:%d)\n", x);

ux = UMAX; // 수정

if (ux > -2)

printf("#1 Passed! (ux:%u)\n", ux);

else

printf("#1 Failed! (ux:%u)\n", ux);

x = 2016124087; // 수정

y = TMIN; // 수정

if (x>y && -x>-y)

printf("#1 Passed! (x,y:%d,%d)\n", x,y);

else

printf("#1 Failed! (x,y:%d,%d)\n", x,y);

x = 2016124087; // 수정

y = -2016124087; // 수정

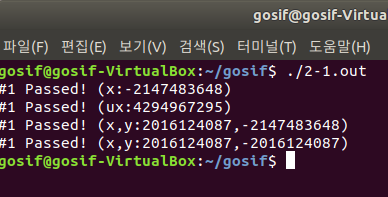
if (x>>3 == x/8 && y>>3 != y/8)

printf("#1 Passed! (x,y:%d,%d)\n", x,y);

else

printf("#1 Failed! (x,y:%d,%d)\n", x,y);

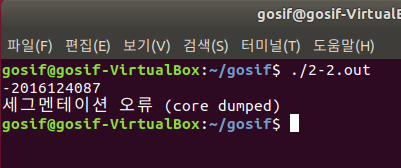
}

A1\_2) 실행 결과  
  
A1\_3) 코드 설명  
- 첫 번째 수정 : x<0을 만족하며 x\*2>=0 이므로 x는 음수이면서 2배 곱하면 양수가 되어야 한다. 논리적으로는 되는 숫자가 없겠지만, overflow를 이용하면 가능하다. TMIN에서 2배했을 때 overflow가 되는 선까지는 부호 비트를 나타내는 최상위 비트 값이 1에서 0으로 바뀌게 되어 양수로 바뀌게 될 것이다.  
- 두 번째 수정 : ux > -2 보다 커야 하는데, 얼핏 보면 모든 변수가 Passed 될 것 같지만, 비교하는 것의 첫번째 타입이 unsigned 형식이므로 -2도 unsigned라고 생각하고 비교하게 된다. 따라서 비트로 보면 -2는 11…10이므로 이는 unsigned 타입의 가장 큰 수인 UMAX(11…11)보다 1작은 수이므로 오로지 UMAX에서만 통과한다.  
- 세 번째 수정 : x>y 이면서 -x>-y 이여야 하므로 수학적으로는 만족하는 x,y가 없다. 하지만, TMIN의 경우에는 10…00인데 이는 양수로 바꾸게 되어도(-를 곱해) 10…00인 것을 알 수 있다. 따라서 y가 TMIN이면 x는 TMIN보다 높은 수이면 Passed를 하게 된다.  
- 네 번째 수정 : x>>3 == x/8에 대해서는 x를 3비트만큼 right shift를 한 것이 x/8과 같아야 하는 것인데, 이는 8=2^3이므로 어떤 x에 대해서도 될 것 같지만, x가 음수인 경우에는 이를 만족하지 못한다. 따라서 x는 양수 값, y는 음수 값을 넣으면 Passed를 하게 된다.

2. 다음 프로그램에 대하여 다음을 수행하시오.

|  |
| --- |
| #include <stdio.h>  #include <stdlib.h>  #include <string.h>  char a[27];  int main() {  char\* b = "abcdefghijklmnopqrstuvwxyz";  int num = 0;  scanf("%d", &num);  if (num > 26)  memcpy(a, b, 26);  else  memcpy(a, b, num);  printf("%s\n", a);  } |

1) 위 프로그램이 비정상적으로 종료하도록 수행시키고 그 원인을 설명하시오.

  
원인 설명 : 프로그램의 흐름을 보면 b라는 char 포인터가 있고 거기에는 “abcdefghijklmnopqrstuvwxyz”가 저장되어 있다. 그리고 num 26이상이면 26만큼, num값을 입력 받아 num값 만큼의 길이의 b의 데이터를 a에 저장하는 함수이다. 이 때 길이 값(size)이 음수이면 말이 안되므로 프로그램 상 예외를 출력하게 된다.

2) 위 오류가 발생하지 않도록 프로그램을 수정하시오.  
오류가 발생하지 않도록 하려면 음수 값을 입력하더라도 num값이 양수 값이 되어야 오류는 발생하지 않는다. 따라서 num 의 타입을 unsigned로 선언해도 되고, 아래와 같이 조건문을 더 추가해도 된다.  
A2\_1) 소스 코드  
#include <stdio.h>

#include <stdlib.h>

#include <string.h>

char a[27];

int main() {

char\* b = "abcdefghijklmnopqrstuvwxyz";

int num = 0;

scanf("%d", &num);

if (num > 26)

memcpy(a, b, 26);

else if (num < -26)

memcpy(a, b, 26);

else if (num < 0) {

num \*= -1;

memcpy(a, b, num); }

else

memcpy(a, b, num);

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

}

A2\_2) 실행 결과  
