**CAT-CERT 2022 C #5 5.4~5.10 과제**

**202221162 김민재**

**목차**

1. 5.4 실습
2. 조사 과제
3. 실습 과제
4. **5.4 실습**

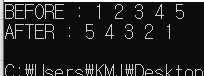
**1. 포인터 1**

|  |
| --- |
| #include <stdio.h>  void swap(int \*a, int \*b) {  int tmp = \*a;  \*a = \*b;  \*b = tmp;  }  int main() {  int a = 1, b = 7;  printf("%d %d\n", a, b);  swap(&a, &b);  printf("%d %d\n", a, b);  } |

****

**2. 포인터 2**

|  |
| --- |
| #include <stdio.h>  void reverse(int\* arr, int len) {  for (int i = 0; i < len / 2; i++) {  int tmp = arr[i];  arr[i] = arr[len - i - 1];  arr[len - i - 1] = tmp;  }  }  int main() {  int arr[5] = { 1,2,3,4,5 };  printf("BEFORE : ");  for (int i = 0; i < 5; i++)  printf("%d ", arr[i]);  reverse(arr, 5);  printf("\nAFTER : ");  for (int i = 0; i < 5; i++)  printf("%d ", arr[i]);  printf("\n");  } |

****

**3. 포인터 3**

|  |
| --- |
| #define \_CRT\_SECURE\_NO\_WARNINGS  #include <stdio.h>  void sort(int\* arr) {  int tmp;  for (int i = 0; i < 9; i++)  for (int j = 0; j < 9 - i; j++)  if (arr[j] < arr[j + 1]) {  tmp = arr[j];  arr[j] = arr[j + 1];  arr[j + 1] = tmp;  }  }  int main() {  int arr[100];  for (int i = 0; i < 10; i++)  scanf("%d", &arr[i]);  sort(arr);  for (int i = 0; i < 10; i++)  printf("%d ", arr[i]);  } |

****

****

1. **조사 과제**

**1. math.h**

여러 수학 함수들을 포함하는 C 언어의 표준 라이브러리이다. 각도는 라디안을 사용한다.

|  |  |
| --- | --- |
| **함수 원형** | **설명** |
| **삼각 함수** | |
| double **sin** ( double x ); | 사인 x를 구한다. |
| double **cos** ( double x ); | 코사인 x를 구한다. |
| double **tan** ( double x ); | 탄젠트 x를 구한다. |
| double **hypot** ( double side1, double side2); | 변 길이가 side1, side2인 직각 삼각형의 빗변을 구한다. |
| **지수 · 대수 함수** | |
| double **exp** ( double x ); | ex를 구한다. |
| double **frexp** ( double x, int \* exp ); | 지수를 exp가 가리키는 변수에 저장하고 가수를 반환한다. |
| double **ldexp** ( double x, int exp ); | x \* 2exp를 반환한다. |
| double **log** ( double x ); | loge x (자연로그)를 구한다. |
| double **log10** ( double x ); | log10 x를 구한다. |
| double **modf** ( double x, double \* intpart ); | 정수부를 intpart가 가리키는 변수에 저장하고 소수부를 반환한다. |
| **거듭제곱 · 거듭제곱근 · 올림 · 내림 · 절댓값 · 나머지 함수** | |
| double **pow** ( double x, double y ); | xy를 구한다. |
| double **sqrt** ( double x ); | {\displaystyle {\sqrt {x}}}x의 제곱근을 구한다. |
| double **ceil** ( double x ); | x보다 작지 않은 가장 작은 정수를 구한다. |
| double **floor** ( double x ); | x보다 크지 않은 가장 큰 정수를 구한다. |
| int **abs** ( int x );  long **labs** ( long x );  double **fabs** (double x); | x의 절댓값을 구한다. |
| double **fmod** ( double x, double y ); | x를 y로 나눈 나머지를 구한다. |

외에 역 삼각 함수, 쌍곡선 함수, 오차 함수, 감마 함수, 퀀텀 함수 등이 있다.

|  |  |
| --- | --- |
| **이름** | **설명** |
| **상수** | |
| HUGE\_VAL | 아주 큰 값을 나타낸다. 수학 계산에서 결과가 너무 커 오버플로우가 나면 이 값을 반환한다. |
| M\_E | 자연상수 e |
| M\_PI | 원주율 π |

**- 예제**

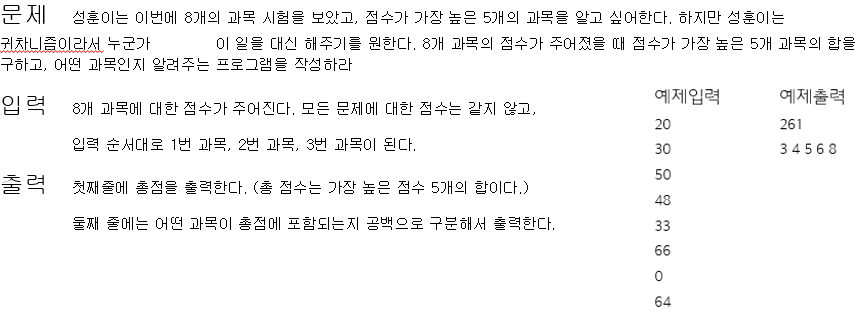
|  |
| --- |
| #include <stdio.h>  #include <math.h>  int main() {  double a = 3.0, b = 4.0, c = 1.234, pi = 3.141592;  printf("a = 3.0\nb = 4.0\nc = 1.234\npi = 3.141592\n\n삼각 함수\nsin(pi/3) = %f\ncos(pi) = %f\ntan(pi/3) = %f\n", sin(pi/3), cos(pi/3), tan(pi/4));  printf("hypot(a, b) = %f\n", hypot(a, b));  printf("\n지수 함수");  printf("\nexp(1) = %f\nexp(a) = %f\nexp(b) = %f", exp(1), exp(a), exp(b));  printf("\nldexp(a, b) = %f", ldexp(a, b));  printf("\nlog(exp(1)) = %f\nlog(exp(a)) = %f", log(exp(1)), log(exp(a)));  printf("\nlog10(c) = %f\nlog10(c \* 10) = %f", log10(c), log10(c \* 10));  printf("\n\n기타 함수");  printf("\npow(a, b) = %f\npow(a, pi) = %f", pow(a, b), pow(a, pi));  printf("\nsqrt(b) = %f", sqrt(b));  printf("\nceil(c) = %f\nfloor(c) = %f", ceil(c), floor(c));  printf("\nlog10(c / 100) = %f\nfabs(log10(c / 100)) = %f", log10(c / 100), fabs(log10(c / 100)));  printf("\nfmod(a, c) = %f", fmod(a, c));  } |

**2. 32bit 와 64bit 의 차이**

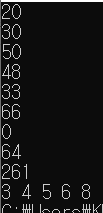
|  |  |  |
| --- | --- | --- |
| **CPU** | **32bit (x86)** | **64bit (x64)** |
| 한번에 처리 가능한 단위 | 32bit (4byte) | 64bit (8byte) |
| 표현 가능한 범위 | 2^32 (4,294,967,296) | 2^64 (18,446,744,073,709,551,616) |
| 처리 속도 | 비교적 느림 | 비교적 빠름 |
| 32bit 운영체제 | O | O |
| 64bit 운영체제 | X | O |

1. **실습 과제**

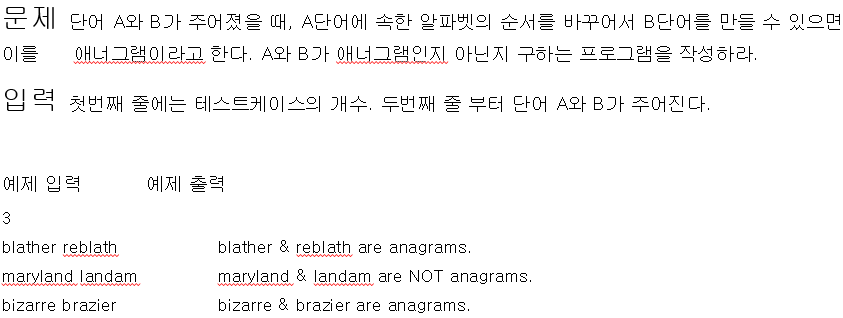
**#1 시험**

****

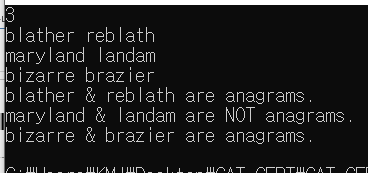
|  |
| --- |
| #define \_CRT\_SECURE\_NO\_WARNINGS  #include <stdio.h>  int main() {  int tmp, score[8], s2[8];  for (int i = 0; i < 8; i++) {  scanf("%d", &score[i]);  s2[i] = score[i];  }  for (int i = 0; i < 8; i++)  for (int j = 0; j < 7 - i; j++)  if (s2[j] < s2[j + 1]) {  tmp = s2[j];  s2[j] = s2[j + 1];  s2[j + 1] = tmp;  }  printf("%d\n", s2[0] + s2[1] + s2[2] + s2[3] + s2[4]);  for (int i = 0; i < 8; i++)  if (score[i] > s2[5])  printf("%d ", i + 1);  } |

****

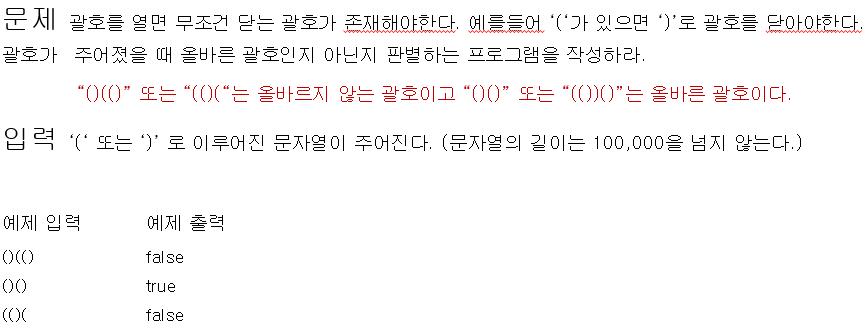
**#2 에너그램**

****

|  |
| --- |
| #define \_CRT\_SECURE\_NO\_WARNINGS  #include <stdio.h>  #include <string.h>  void scanalp(char \*word, int \*alp) {  for (int i = 0; word[i]; i++)  alp[word[i] - 97]++;  }  int main() {  char word[10001], answer[1000000] = "", check;  int dest[26], source[26], count;  scanf("%d", &count);  for (int i = 0; i < count; i++) {  memset(dest, 0, sizeof(int) \* 26);  memset(source, 0, sizeof(int) \* 26);  scanf("%10000s", word);  scanalp(word, dest);  strcat(answer, word);  strcat(answer, " & ");  scanf("%10000s", word);  scanalp(word, source);  strcat(answer, word);  check = 0;  for (int j = 0; j < 26; j++)  if (dest[j] != source[j]) {  check++;  break;  }  strcat(answer, (check) ? " are NOT anagrams.\n" : " are anagrams.\n");  }  printf("%s", answer);  } |

****

**#3 괄호**

****

|  |
| --- |
| #define \_CRT\_SECURE\_NO\_WARNINGS  #include <stdio.h>  int main() {  char str[100001];  int open = 0;  scanf("%100000s", str);  for (int i = 0; str[i]; i++) {  if (str[i] == '(')  open++;  else  open--;  if (open < 0) break;  }  printf("%s", (open) ? "false" : "true");  } |

****

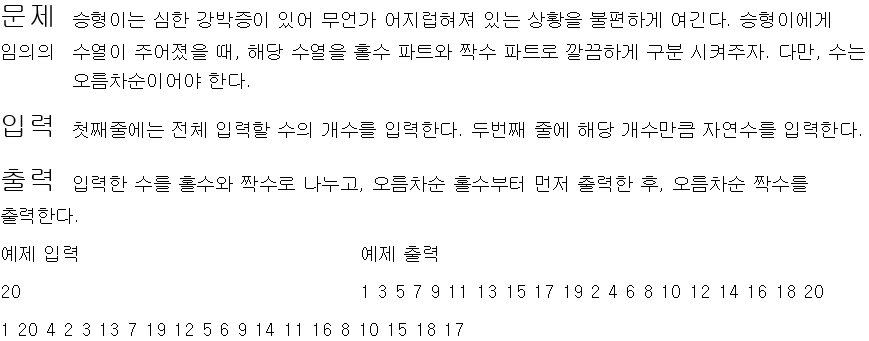
****

****

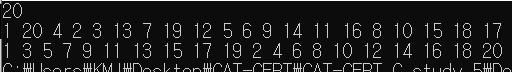
****

****

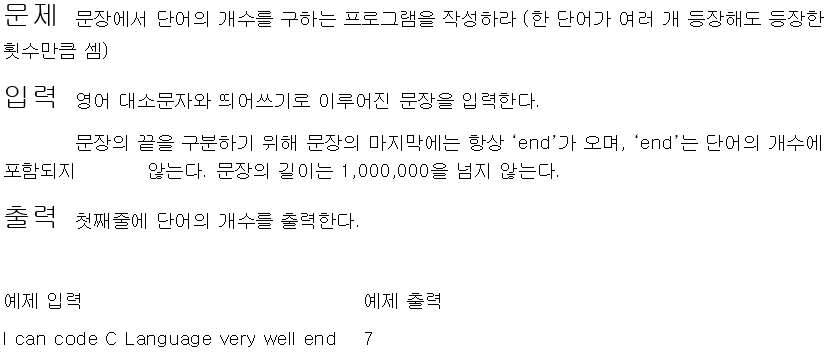
**#4 강박증**

****

|  |
| --- |
| #define \_CRT\_SECURE\_NO\_WARNINGS  #include <stdio.h>  int main() {  int count, tmp, arr[100000] = { 0 };  scanf("%d", &count);  for (int i = 0; i < count; i++)  scanf("%d", &arr[i]);  for (int i = 1; i < count; i++)  for (int j = 0; j < count - i; j++)  if (arr[j] % 2 < arr[j + 1] % 2 || arr[j] % 2 == arr[j + 1] % 2 && arr[j] > arr[j + 1]) {  tmp = arr[j];  arr[j] = arr[j + 1];  arr[j + 1] = tmp;  }  for (int i = 0; i < count; i++)  printf("%d ", arr[i]);  } |

****

**#5 단어의 개수**

****

|  |
| --- |
| #define \_CRT\_SECURE\_NO\_WARNINGS  #include <stdio.h>  int main() {  char word[1000001];  int count = 0;  scanf("%1000000[^\n]s", word);  for (int i = 0; word[i]; i++) {  if (word[i] == ' ') count++;  else if (word[i] == 'e') if (word[i + 1] == 'n') if (word[i + 2] == 'd') break;  }  printf("%d", count);  } |

****