

IT로 수학문제를 해결해 보자

프로그래밍 2차 수행평가

문제1

골드바흐의 추측 : 2보다 큰 모든 짝수는 2개의 소수의 합으로 표시할 수 있다. 이때, 1개의 소수를 2번 사용하는 것을 허용한다.

짝수 2,147,438,646 은 4바이트 int형으로 표현할 수 있는 가장 큰 짝수이다. 이 수를 두 소수의 합으로 표시할 수 있는 방법은 몇 가지가 있는지 찾아 보자!

문제해결을 위한 알고리즘의 순서도를 그리고, 주요 연산을 파악한 후 연산 횟수를 계산하고, C언어로 구현한 후 결과를 도출해 보세요. 또한 연산 횟수를 줄일 수 있는 방법에 대해 고민해 보세요.

문제2

팬디지털 : 1부터 n 까지 숫자를 하나씩만 사용하여 만든 n 자리 숫자 예) 2143 : 4자리 팬디지털인 동시에 소수

N 자리 팬디지털 소수 중 가장 큰 수가 무엇인지 찾아보자.

문제해결을 위한 알고리즘의 순서도를 그리고, 주요 연산을 파악한 후 연산 횟수를 계산하고, C언어로 구현한 후 결과를 도출해 보세요. 또한 연산 횟수를 줄일 수 있는 방법에 대해 고민해 보세요.

문제3

$39 \times 186 = 7254$ 이 곱셈식은 1~9까지 숫자를 1번씩 사용해서 만든 곱셈식이다.

위의 예와 같이 1~9까지 숫자를 한번씩만 사용해서 $a \times b = c$ 라는 곱셈식을 만족하는 c 의 개수를 몇 개인지 찾아보자.

문제해결을 위한 알고리즘의 순서도를 그리고, 주요 연산을 파악한 후 연산 횟수를 계산하고, C언어로 구현한 후 결과를 도출해 보세요. 또한 연산 횟수를 줄일 수 있는 방법에 대해 고민해 보세요.

문제4

순환소수 : 소수 중에서 각 자리의 숫자들을 순환시켜도 여전히 소수인 수

예) 197은 971, 719 모두 소수이므로 순환 소수이다.

100,000이하의 순환소수는 몇 개인지를 찾아보자.

문제해결을 위한 알고리즘의 순서도를 그리고, 주요 연산을 파악한 후 연산 횟수를 계산하고, C언어로 구현한 후 결과를 도출해 보세요. 또한 연산 횟수를 줄일 수 있는 방법에 대해 고민해 보세요.

문제5

소수 3137은 왼쪽에서 하나씩 숫자를 제거했을 때 만들어지는 수의 집합 $\{3137, 137, 37, 7\}$ 과 오른쪽에서 하나씩 숫자를 제거했을 때 만들어지는 수의 집합 $\{3137, 313, 31, 3\}$ 이 모두 소수로 이루어져 있다. 이런 특성을 가지는 소수는 2,3,5,7을 제외하고 총 11개 존재한다고 한다.

이러한 특성을 가지는 11개의 소수를 찾아보자.

문제해결을 위한 알고리즘의 순서도를 그리고, 주요 연산을 파악한 후 연산 횟수를 계산하고, C언어로 구현한 후 결과를 도출해 보세요. 또한 연산 횟수를 줄일 수 있는 방법에 대해 고민해 보세요.

문제6

교실에 1~2000번까지 번호가 붙여진 사물함이 있다. 처음에 모든 사물함의 문은 닫혀 있다. 학생은 1번부터 1000번까지 1000명이며, 모두 아래와 같은 작업을 번호 순서대로 진행한다.

<아래>

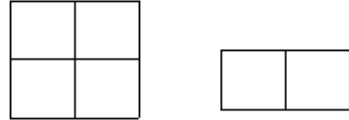
자기 번호의 배수가 되는 모든 번호의 사물함에 대하여 사물함이 닫혀 있으면 열고, 열려 있으면 닫는다.

1번~1000번까지 모두 작업을 한 후 마지막에 열려 있는 사물함은 모두 몇 개인지 찾아보자.

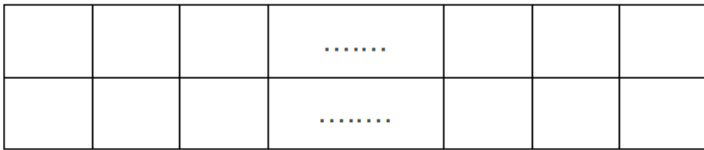
문제해결을 위한 알고리즘의 순서도를 그리고, 주요 연산을 파악한 후 연산 횟수를 계산하고, C언어로 구현한 후 결과를 도출해 보세요. 또한 연산 횟수를 줄일 수 있는 방법에 대해 고민해 보세요.

문제7

다음과 같이 2가지 종류의 블록이 있다.



이 블록들을 여러 개 사용해서 $2 \times k$ 모양의 판을 채우는 경우의 수를 구하고자 한다.



가로 길이가 k일 때의 경우의 수를 구하는 일반 식을 도출하고, k가 10일 때의 값을 구해보자.

문제해결을 위한 알고리즘의 순서도를 그리고, 주요 연산을 파악한 후 연산 횟수를 계산하고, C언어로 구현한 후 결과를 도출해 보세요. 또한 연산 횟수를 줄일 수 있는 방법에 대해 고민해 보세요.

문제8

숫자 512는 각 자릿수를 모두 합한 다음 거듭제곱을 해보면 자신이 된다. $5 + 1 + 2 = 8$ 인데 $8 * 8 * 8 = 512$ 이다.

이런 숫자들을 크기 순으로 나열한 수열의 n 번째 항을 a_n 이라고 할 때, $a_2 = 512$ 이다. (1자리 숫자는 제외한다.)

a_{11} 번째 항은 어떤 수인지 찾아 보자.

문제해결을 위한 알고리즘의 순서도를 그리고, 주요 연산을 파악한 후 연산 횟수를 계산하고, C언어로 구현한 후 결과를 도출해 보세요. 또한 연산 횟수를 줄일 수 있는 방법에 대해 고민해 보세요.

문제9

$P(N)$ 은 1부터 N 까지의 숫자를 나열하고 숫자를 아래의 규칙대로 삭제해 나갔을 때, 마지막에 남은 숫자로 정의한다.

<규칙>

- 1) 첫번째 숫자를 삭제한 후, 숫자 하나를 건너뛰면서 삭제해 나간다.
- 2) 숫자 나열의 마지막까지 갔으면 반대 방향으로 1번을 반복한다.

예를 들어 $N = 9$ 이면

1 2 3 4 5 6 7 8 9 에서 1을 삭제하고, 3, 5, 7, 9를 삭제한다. 2, 4, 6, 8 이 남은 상태에서 반대 방향으로 8을 삭제하고 4를 삭제한다. 2, 6이 남은 상태에서 2를 삭제한다. 따라서 $P(9) = 6$ 이다.

$S(N)$ 은 $P(1)$ 부터 $P(N)$ 까지의 합일 때, $S(1000)$ 을 구해 보자. $P(1) = 1$ 이다.

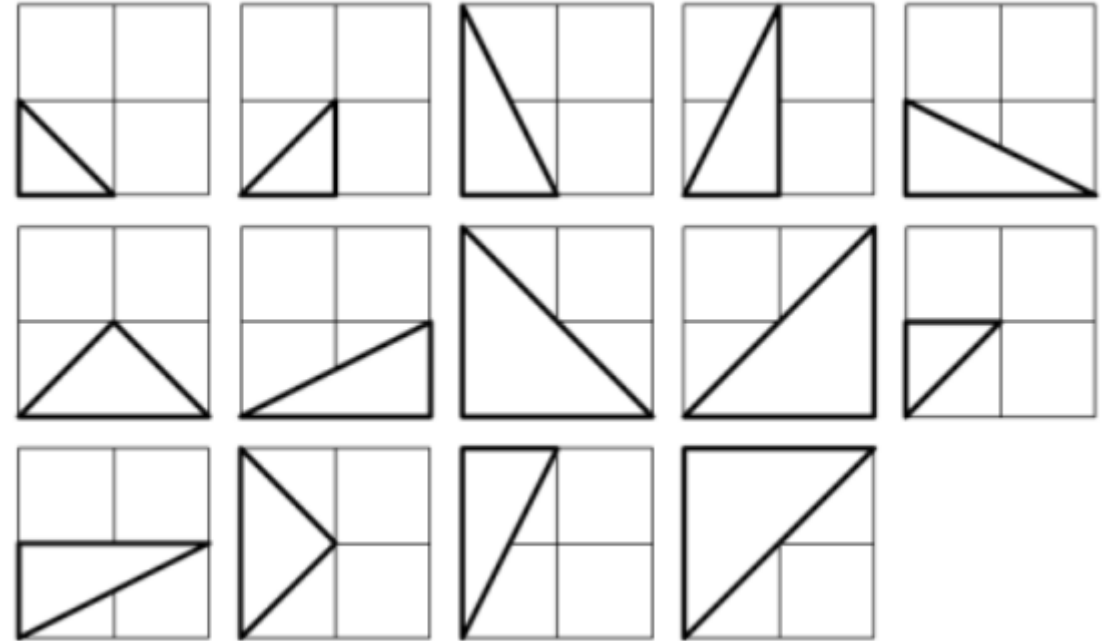
문제해결을 위한 알고리즘의 순서도를 그리고, 주요 연산을 파악한 후 연산 횟수를 계산하고, C언어로 구현한 후 결과를 도출해 보세요. 또한 연산 횟수를 줄일 수 있는 방법에 대해 고민해 보세요.

문제10

정수 좌표계 상에서 두 점 $P(x_1, y_1)$ 와 $Q(x_2, y_2)$ 를 잡고 원점 $O(0,0)$ 와 이르면 삼각형 $\triangle OPQ$ 가 만들어 진다.

x, y 좌표값이 0과 2사이일 때, 만들 수 있는 직각 삼각형은 모두 14개이다.

x, y 좌표값의 범위가 0~20일 때, 만들 수 있는 직각 삼각형은 모두 몇 개인가?



문제해결을 위한 알고리즘의 순서도를 그리고, 주요 연산을 파악한 후 연산 횟수를 계산하고, C언어로 구현한 후 결과를 도출해 보세요. 또한 연산 횟수를 줄일 수 있는 방법에 대해 고민해 보세요.

문제11

1234321234321234321234321.....

위와 같이 반복되는 숫자 나열이 있다.

이 숫자 나열을 적절히 잘랐을 때, 숫자의 합이 항의 위치가 된다.

예를 들어 1, 2, 3, 4, 32, 123, 43, 2123, 432, 1234,

이렇게 자르면 그 자릿수의 합은 1,2,3,4,5,6,7,8,9,10,이다.

V_n 이 n 번째 수열의 값이라면 $V_2 = 2$, $V_5 = 32$ 이다.

이때, $S(n) = V_1 + V_2 + \dots + V_n$ 이면 $S(11) = 36120$ 이다.

$S(1000)$ 을 123454321 로 나눈 나머지를 구해 보자.

문제해결을 위한 알고리즘의 순서도를 그리고, 주요 연산을 파악한 후 연산 횟수를 계산하고, C언어로 구현한 후 결과를 도출해 보세요. 또한 연산 횟수를 줄일 수 있는 방법에 대해 고민해 보세요.

문제12

모든 양의 정수 n 에 대해서 $S(n) = k$ 는 $n+k$ 가 $k+1$ 로 나누어 떨어지지 않는 가장 작은 수 k 로 정의한다. 예를 들어 $S(13) = 4$ 이며, $S(120) = 1$ 이다.

$1 < n < 14$ 범위에서 $S(n) = 3$ 을 만족하는 정수 n 의 개수는 1개이다.

$1 < n < 1,000,000$ 사이에서 $S(n) = i$ ($1 < i < 10$) 을 만족하는 n 의 개수의 합을 구해보자.

문제해결을 위한 알고리즘의 순서도를 그리고, 주요 연산을 파악한 후 연산 횟수를 계산하고, C언어로 구현한 후 결과를 도출해 보세요. 또한 연산 횟수를 줄일 수 있는 방법에 대해 고민해 보세요.

문제13

아래의 조건을 만족하는 (a,b,c) 의 합을 $S(n) = \sum(a+b+c)$ 으로 정의한다.

<아래>

- 1) a, b, c 는 소수이다.
- 2) $a < b < c < n$
- 3) $(a + 1) * k = (b+1), (b+1) * k = (c+1) : k$ 는 임의의 수

예를 들어 $S(12)$ 를 만족하는 것은 $(2,5,11)$ 밖에 없으며 $S(12) = 18$ 이다.

$S(1,000)$ 를 구해 보자.

문제해결을 위한 알고리즘의 순서도를 그리고, 주요 연산을 파악한 후 연산 횟수를 계산하고, C언어로 구현한 후 결과를 도출해 보세요. 또한 연산 횟수를 줄일 수 있는 방법에 대해 고민해 보세요.

문제14

$A^2 \% 6$ ($0 \leq a \leq 5$)의 값은 0, 1, 4, 3, 2, 1로 가장 큰 값은 4이다. $F(n)$ 은 $0 \leq a < n$ 범위의 a^2 을 6으로 나누었을 때의 나머지 중 가장 큰 수를 계산한다고 하면, $F(6) = 4$ 라 할 수 있다.

이때 $1 \leq n \leq 105$ 일때, $F(n)$ 의 합을 구해 보자.

문제해결을 위한 알고리즘의 순서도를 그리고, 주요 연산을 파악한 후 연산 횟수를 계산하고, C언어로 구현한 후 결과를 도출해 보세요. 또한 연산 횟수를 줄일 수 있는 방법에 대해 고민해 보세요.

문제15

1~ ∞ 번호표를 가진 사람들이 기차의 좌석 배정을 받기 위해 줄 서 있다. 처음엔 기차가 텅 비어있으며, 아래의 규칙에 따라 가장 작은 숫자인 호차에 처음으로 비어있는 좌석 번호를 배정하는 것을 원칙으로 한다. 기차의 호차 번호와 좌석 번호는 무한하다.

<규칙>

- 호차가 비어 있으면 좌석 번호 1번을 배정받는다.
- 직전에 그 호차에 탑승한 사람의 번호와 자신의 번호를 더했을 때, 완전제곱수이면 그 호차의 다음 번호를 배정받는다.

규칙대로 하면 1번인 사람은 1호차의 1번 좌석을 배정받고, 2번인 사람은 $1+2 = 3$ 은 완전제곱수가 아니므로 1호차에 배정받지 못하고 비어있는 2호차의 1번 좌석을 배정받는다. 3번인 사람은 $1 + 3 = 4$ 는 완전제곱수이므로 1호차의 2번 좌석을 배정받는다.

이때, $T(v, s)$ 는 v 번 호차에 s 좌석 번호에 앉은 사람의 번호를 나타낸다. $T(1,1) = 1$, $T(1,2) = 3$, $T(2, 1) = 2$, $T(10, 20)$ 은 440이다. $T(25, 75) + T(99,100)$ 의 계산 결과를 구해 보자

문제해결을 위한 알고리즘의 순서도를 그리고, 주요 연산을 파악한 후 연산 횟수를 계산하고, C언어로 구현한 후 결과를 도출해 보세요. 또한 연산 횟수를 줄일 수 있는 방법에 대해 고민해 보세요.

문제16

2개의 양수 A, B에 대하여 A--B는 아래와 같이 정의한다.

- 1) A와 B의 길이가 같고 1자리 숫자만 다른 경우 ex) 123--173
- 2) A나 B에 왼쪽에 1자리 숫자를 추가하면 서로 같아지는 경우 ex) 23--223 , 123--23

소수 P가 2와 P 사이를 서로 -- 로 이을 수 있고, 2에서 P사이에 P보다 큰 소수가 없을 때, **2→P**로 표시한다.

예를 들어, 2→127은 성립한다.

2--3--13--113--103--107--127, 그러나 11은 2--3--13--11 로 2와 11사이에 11보다 큰 수 13이 있으므로 2→11은 성립하지 않는다.

S(N)은 N보다 작은 소수 P에 대하여 2→P를 만족하지 않는 소수들의 합이다.

S(1000) + S(10000)을 구해 보자!

문제해결을 위한 알고리즘의 순서도를 그리고, 주요 연산을 파악한 후 연산 횟수를 계산하고, C언어로 구현한 후 결과를 도출해 보세요. 또한 연산 횟수를 줄일 수 있는 방법에 대해 고민해 보세요.

문제17

N-Number는 2~N진수로 숫자를 표현했을 때, 0~N-1까지의 숫자를 적어도 한번씩 사용하여 표현할 수 있는 숫자를 의미한다.

예를 들어, 978은 $1111010010_2 = 1100020_3 = 33102_4 = 12403_5$ 로 표현할 수 있으므로 5-Number라 할 수 있다.

10-Number로 가장 작은 수는 1093265784이다. 10-Number중 가장 작은 순서대로 5개의 합을 구해 보자.

문제해결을 위한 알고리즘의 순서도를 그리고, 주요 연산을 파악한 후 연산 횟수를 계산하고, C언어로 구현한 후 결과를 도출해 보세요. 또한 연산 횟수를 줄일 수 있는 방법에 대해 고민해 보세요.

평가 세부 항목

작성 수준	보고서 작성	프로그래밍	정답여부	발표	합계
상	9	5	3	3	
중	7	3	2	2	
하	5	1	1	1	
제출x	1	0	0	0	1

1. 보고서와 C파일을 2018년 6월 17일 23:00 까지 제출해 주세요.
2. 팀원 중 한 명만 제출하면 됩니다.
3. C파일에는 주석을 반드시 달아주세요.
4. 발표는 5분 이내로 준비하시면 됩니다.
5. 제출 기한을 넘기면 1일당 10%감점