

問1. 本講義の内容に関して、次の文章の空欄 (a) ~ (H) を埋めよ (キーワードに関しては順不同)。

- (a) (人名) 型計算機は、ストアードプログラム型計算機ともよばれるが、その特徴は (b) である。
- 10 進数の 1453 を 2 進数に直すと (c) であり、8 進数に直すと (d) であり、16 進数に直すと (e) である。
- 情報とは (f) であり、計算機とは (g) である (本講義で説明した内容以外は不可)。
- 1 bit は (h) を表し、1 byte は通常の場合 (i) を表す。
- 一般に、最も低級なプログラミング言語は (j) である。高級な言語で書かれたプログラムを、低級な言語のプログラムに変換するプログラムを (k) と呼ぶ。
- 計算機におけるメモリーは、CPU などが (l) と、(m) が (n) を指定して、その指定されたデータに対して指定された動作を行う。現在の計算機における通常のメモリーでは、1 つの (l) に対応するデータ量は (o) である。
- Java のキーワードの例として、(p)、(q)、(r)、(s)、(t)、(u)、(v)、(w)、(x)、(y) などがある (キーワードでないものを書くとは減点)。識別子のために使うことができる文字は、アルファベット、(z)、記号の (A) と _ であり、文字数の制限はないが (z) から始まってはいけない。
- int 型の変数は、(B) から (C) までの整数を表すことができる (2^n という表現を使ってよい)。short 型の変数に、データとして FFC3 H が格納されている場合、その変数が表す数は、10 進数では (D) となる。同様に 00C3 H の場合、(E) となる。浮動小数点数、 -2.3×10^4 を、10 のべき乗を残したまま float 型のリテラルで表すと (F) となり、double 型のリテラルで表すと (G) となる。
- float 型の内部データの 16 進表現が C0A80000 H である数を 10 進数で表すと、(H) となる。

問2. 次のプログラム (クラス Chukan2) を実行したときに出力されるものを記せ。

```
public class Chukan2 {
    public static void main(String[] args) {
        double x = 0.0, y = 3.0;
        if (x > 0.0) System.out.println(":-)");
        else if (y > 2.0) System.out.println(":*)");
        else System.out.println(":-D");
        if (x > 0.0 && (y = 2.0) > 1.0) System.out.println("8-)");
        System.out.println("y = " + y);
        if (x > 0.0 & (y = 2.0) > 1.0) System.out.println("B:-)");
        System.out.println("y = " + y);

        char c = '山';
        switch(c) {
            case '山': System.out.println("c は' 山' かな?");
            case '下': System.out.println("k は' 川' かな?"); break;
            case '幸': System.out.println("k は' 幸' かな?");
            case '彦': System.out.println("k は' 彦' かな?"); break;
            default: System.out.println("わかんないや?");
        }

        int l, m, n;
        l = 5;
        m = --l;
        n = l++;
    }
}
```

(プログラムは次ページに続く)

```

    System.out.println("l = " + l + " m = " + m + " n = " + n);
}
}

```

問3 . 次の式の演算子が評価される順序を , 各式の中の全ての演算子の下に 1 から順番に番号付して記せ。

```

a = b + c;
a = b * c / d;
a = b - c / d - e;
a = b + c - d * e;
a = b = c = d + e - f + g;
a = b * c - - ++ d * e * f + g;
A = a == b && c < d || e > f;

```

問4 .

次のプログラムは , キーボードから入力した数が素数かどうか判定するプログラムである。偶数は最初に判定する , 割る数を調べる数の平方根までとするなどして , 多少高速化してある。その空欄 (a) ~ (e) を埋め , プログラムのあとの設問に答えよ。

```

1: import java.util.Scanner;
2: public class Chukan4 {
3:     public static void main(String[] args) {
4:         Scanner stdIn = new Scanner(System.in);
5:         System.out.println("素数かどうか判定する数 (2 以上) を入力して下さい。");
6:         int n = stdIn.nextInt();
7:
8:         if (n % 2 _____ (a) _____ 0) System.out.println(n + "は2で割れるため素数ではない。");
9:         else {
10:             int divisor = 3;
11:             while (divisor * divisor <= n) {
12:                 if (_____ (b) _____) {
13:                     System.out.println(n + "は" + divisor + "で割れるため素数ではない。");
14:                     _____ (c) _____
15:                 }
16:                 divisor _____ (d) _____ 2;
17:             }
18:             if (_____ (e) _____) {
19:                 System.out.println(n + "は素数である。");
20:             }
21:         }
22:     }

```

11 行目の条件式 `divisor * divisor <= n` は , `divisor` が `n` の平方根を越えないためのものであるが , 毎回 2 乗を計算することは効率が悪い。そこで , 最初に 2 分法により `n` の平方根以上の最も小さな整数を計算し (`sqrt` とする) , 条件式を `divisor <= sqrt` とすることを考えた。そのときのプログラムで追加・変更する部分を記述せよ。

なお , `n` の平方根が `lb` 以上 `ub` 以下と分かっている場合 , この場合の 2 分法は , `chk = (lb + ub) / 2` とおいて , `chk` の 2 乗が `n` より小さければ `lb` に `chk` を代入し , 大きいとか等しければ `ub` に `chk` を代入し , `ub - lb` が 1 以下になるまで繰り返し , `ub` を求める平方根として出力する方法である。ただし , 変数が `int` 型であるため , 小数点以下の四捨五入に注意してプログラムを記述すること。

解答用紙

学科・類： 学籍番号： 名前：

問 1 . ((p) ~ (y), (F), (G) は各 1 点 , それ以外各 2 点 , 計 56 点)

(a)	(k)	(w)
(b)	(l)	(x)
	(m)	(y)
	(n)	(z)
(c)	(o)	(A)
(d)	(p)	(B)
(e)	(q)	(C)
(f)	(r)	(D)
(g)	(s)	(E)
(h)	(t)	(F)
(i)	(u)	(G)
(j)	(v)	(H)

問 2 . (10 点)

問 3 . (2 × 7 = 14 点)

$$a = b + c;$$

$$a = b * c / d;$$

$$a = b - c / d - e;$$

$$a = b + c - d * e;$$

$$a = b = c = d + e - f + g;$$

$$a = b * c - - ++ d * e * f + g;$$

$$A = a == b \ \&\& \ c < d \ || \ e > f;$$

問 4 . ($3 \times 5 + 5 = 20$ 点 ((a) +3 点))

- (a)
- (b)
- (c)
- (d)
- (e)

あらかじめ平方根を計算して高速化するプログラムのための変更点

解答用紙

学科・類：

学籍番号：

名前：

問 1 . ((p) ~ (y), (F), (G) は各 1 点, それ以外各 2 点, 計 56 点)

(a) ノイマン

(b) プログラムとなる命令列をメモリーに格納し, その命令を順次取り出し, その命令に従って処理を実行する計算機

(c) $10110101101_{(2)}$

(d) $2655_{(8)}$

(e) 5AD H

(f) 不確実性を解消するもの

(g) 情報を加工する機械

(h) 0 または 1 のような 2 者択一の情報量

(i) 8 bit

(j) 機械語

(k) コンパイラー

(l) アドレス

(m) 読み込み

(n) 書き込み

(o) 1 byte

(p) byte

(q) short

(r) int

(s) long

(t) float

(u) double

(v) char

(w) if

(x) else

(y) for (など)

(z) 数字

(A) \$

(B) -2^{31}

(C) $2^{31} - 1$

(D) -61

(E) 195

(F) -2.3e4f

(G)) -2.3e4

(H) -5.25

問 2 . (10 点)

```

:*)
y = 3.0
y = 2.0
c は' 山' かな?
k は' 川' かな?
l = 5 m = 4 n = 4
    
```

問 3 . ($2 \times 7 = 14$ 点)

```

a  =  b  +  c;
  2    1
    
```

```

a  =  b  *  c  /  d;
  3    1    2
    
```

```

a  =  b  -  c  /  d  -  e;
  4    2    1    3
    
```

```

a  =  b  +  c  -  d  *  e;
  4    1    3    2
    
```

```

a  =  b  =  c  =  d  +  e  -  f  +  g;
  6    5    4    1    2    3
    
```

```

a  =  b  *  c  -  -  ++  d  *  e  *  f  +  g;
  8    1    6  3  2    4    5    7
    
```

```

A  =  a  ==  b  &&  c  <  d  ||  e  >  f;
  6    1    3    2    5    4
    
```

問4 . ($3 \times 5 + 5 = 20$ 点 ((a) +3 点))

(a) `== 0 && n - 2 !=`

(b) `n % divisor == 0`

(c) `break;`

(d) `+=`

(e) `divisor * divisor > n`

あらかじめ平方根を計算して高速化するプログラムのための変更点

9 行目の 10 行目の間に追加

```
int ub = n / 3, lb = 3, chk = 3, sqrt;
while(true) {
    chk = (ub + lb) / 2;
    if(chk * chk < n) lb = chk;
    else ub = chk;
    if (ub - lb <= 1) {
        chk = ub;
        break;
    }
}
sqrt = chk;
```

11 行目の変更

```
while (divisor <= sqrt) {
```

18 行目の変更

```
if (divisor > sqrt) {
```