

学籍番号 _____ 氏名 _____

1. UNIX のコマンド

1.1 ファイル(各1点)

- (1) `/D1/D3/f1`
- (2) `../../../../D1/D3/f1`

1.2 ファイル(各1点)

- (1) `.`
- (2) `..`
- (3) `/`

1.3 コマンド(各1点)

- | | |
|-----------------------------|-----------------------------------|
| (1) <code>pwd</code> | (2) <code>ls</code> |
| (3) <code>cd ..</code> | (4) <code>cd hoge</code> |
| (5) <code>cd</code> | (6) <code>mkdir hoge</code> |
| (7) <code>rmdir hoge</code> | (8) <code>rm -rf hoge</code> |
| (9) <code>rm hoge</code> | (10) <code>mv hoge ../huga</code> |

1.4 コンパイル・実行(1:3点 2:各2点)

- (1) 次の例のように、ファイル名の最後に、`.c` が必要である。

`sample.c`

- (2)
- [ア] `cc -lm -o testrun test.c`
- [イ] `cc -o testrun test.c`

2. C 言語の文法

2.1 基礎(各3点)

- (1) 書きたいコメント文を `/*` と `*/` で囲む。
- (2) プログラムの内容をわかりやすくするために書く。

2.2 変数(1:3点 2:各2点 3:各2点)

(1) ローカル変数は、関数の中で定義され、その関数の中だけで使用できる。一方、グローバル変数は、関数の外で定義され、どの関数からでも使用できる。

(2)

[ア] `x`

[イ] `y, theta, a, b`

(3)

[ア] `char aa, bb;`

[イ] `int i, j, k, l, m;`

[ウ] `double A, E, F;`

2.3 制御文(各3点)

(1)

```
if(i<=100){
    y=sin(x);
    i=1;
}else{
    y=cos(x);
    i=-1;
}
```

(2)

```
for(i=100;i<=100;i++){
    sum+=i;
}
```

(3)

```
while(i<=100){
    sum+=i;
    i++;
}
```

2.4 配列(1:3点 2:各2点)

(1) 変数の数が多い場合、宣言が簡単である。さらに、変数の値を操作するときに、添え字の数字で指定できるので、大量のデータを処理するのに向いている。

(2)

[ア] `int a[101][101]`

[イ] `double a[11][51][101]`

2.5 関数(各2点)

(1) `main f g`

(1) `main → f → g`

2.6 ファイル処理(各2点)

[ア] `FILE`

[イ] `fopen`

[ウ] `calresult`

[エ] `fprintf`

[オ] `fclose`

3. プログラムの作成

3.1 メッセージの表示 (各 15 点)

```
#include <stdio.h>

int main()
{
    printf("Hello world\n");
    printf("Hello Akita");

    return(0);
}
```

3.2 関数の利用 (各 7 点)

```
#include <stdio.h>
#include <math.h>

double f(double x);
double g(double x);

int main(){
    int i, n=100;
    double x, pi=4.0*atan(1.0);

    for(i=0;i<=n;i++){
        x=pi*i/n;
        printf("%lf\t%lf\t%lf\n", x, f(x), g(x));
    }

    return(0);
}

double f(double x){
    return(x-pow(x,3)/6.0+pow(x,5)/120-pow(x,7)/5040);
}

double g(double x){
    return(1-pow(x,2)/2.0+pow(x,3)/24-pow(x,6)/720);
}
```

3.3 ネピア数の計算とファイル出力 (各 7 点)

```
#include <stdio.h>
int main(){
    int i, j;
    double e, x;
    FILE *out;

    out=fopen("result.txt", "w");
    e=1.0;
    x=1.0;

    for(i=1;i<=20;i++){
        if(i != 1){
            x/= (i-1);
            e+=x;
        }

        printf("%d%\t%.20f\n", i, e);
        fprintf(out, "%d%\t%.20f\n", i, e);
    }
    fclose(out);

    return(0);
}
```