

学籍番号 _____ 氏名 _____

1. 基礎(各1点)

1.1 コンピューター

[ア] ハードウェア [イ] プログラム [ウ] ソフトウェア [エ] マシン語 [オ] アルゴリズム

1.2 プログラムの作成

[ア] コーディング [イ] ソースプログラム [ウ] コンパイル [エ] リンク [オ] コンパイラー
[カ] リンカー [キ] 実行プログラム [ク] 実行ファイル [ケ] 実行ファイル名 [コ] デバッグ

2. UNIXのコマンド(各1点)

2.1 ファイル

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

2.2 コマンド

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

2.3 コンパイル・実行

- (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) cc -o testrun test.c
- (3) ターミナル上で、testrun とタイプした後、[enter] キーを押す。

3. C 言語の文法(各3点)

3.1 基礎

- (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。

3.2 printf() 関数

- (1) ディスプレイに文字や数値を書き出す。
- (2) printf 関数のダブルクォーテーションで囲まれた部分で、改行したい位置に、\n を書く。
- (3) `printf("i=%5d\n", i);`
- (4) `printf("x=%.5f\n", x);`

3.3 変数

- (1) 変数は、値 (数値や文字など) を記憶するために用いる。
- (2) `int i, j;`
- (3) `double x, y;`
- (4) `z=1.234;`

3.4 scanf() 関数

- (1) キーボードから値 (数値や文字など) を読み込み、変数に格納する。
- (2) `scanf("%d", &i);`
- (3) `scanf("%lf", &x);`
- (4) `z=1.234;`

4. プログラムの作成 (各 14 点)

4.1 メッセージの表示

```
#include <stdio.h>

int main()
{
    printf("Hello.\n\n");
    printf("");

    return(0);
}
```

4.2 携帯電話料金計算プログラム

```
#include <stdio.h>

int main()
{
    int base, time, number;

    base = 3000;

    printf("通話時間を入力してください[分単位]\n");
    scanf("%d", &time);

    printf("メール送信数を入力してください\n");
    scanf("%d", &number);

    printf("----- 使用料金 -----\n");
    printf("%d 円\n", base+10*time+5*number);

    return(0);
}
```