

令和4年度

生活環境学部

第3年次編入学者選抜学力試験問題

小 論 文

〔情報衣環境学科 生活情報通信科学コース〕

令和3年6月12日(土)

13:00～14:30

注 意

1. 解答は、別添の解答用紙（2枚同封）を使用し、問題ごとに別の解答用紙を用いること。
2. 総ページ数 ——— 3ページ
問題ページ ——— 第2～3ページ
（第1ページは、下書き用紙）
3. 試験終了後、この問題冊子は持ち帰ること。

問題 1. プログラム言語 C または Java で、標準入力から二つの整数値を入力しその和を標準出力に出力するプログラムを書け。そのプログラムで使っている C または Java の標準ライブラリ（その言語で標準的に提供されている関数、クラス等）を列挙し、その動作と使い方について詳細に説明せよ。

必要ならばプログラムの他に数式や図を用いてもよい。C にも Java にもなじみがなければ他の言語での同等のプログラムを書きそこで使っている標準ライブラリを説明しても良いが、何らかの意味で標準規格のある言語に限る。

問題 2. 線形探索などを高速化する技法として番兵と呼ばれるものが知られている。これは、探索したい値を列の末端等に事前に配置することで、添字の比較回数を減らすものである。たとえば、図 1 は通常の線形探索アルゴリズムをプログラミング言語 C の関数定義として実装したものであるが、それに手を入れて番兵を使うように変更したものが図 2 である。いずれも、int の配列から特定の値を持つ要素を探索し、あればその要素を指すポインタ（複数ある場合は添字が最小のもの）を返し、なければ NULL を返す。

番兵による高速化の効果と限界について論ぜよ。ただし、以下の点に留意せよ。

- 図 1 の実装と比べて図 2 の実装はどれくらい速くなるか。具体的な入力例での検討も含むことが望ましい。

図 1 通常の線形探索

```
int *
linear_search(int a[], size_t n, int key)
{
    size_t i;
    for (i = 0; i < n; i++) {
        if (a[i] == key)
            return &a[i];
    }
    return NULL;
}
```

図 2 番兵を使う線形探索

```
int *
linear_search(int a[], size_t n, int key)
{
    size_t i;
    a[n] = key;
    for (i = 0; ; i++) {
        if (a[i] == key) {
            break;
        }
    }
    return i < n ? &a[i] : NULL;
}
```
