

新潟大学工学部

プログラム	プログラム (コース)
受験番号	

※合計点	
------	--

チェック欄

※この試験科目を解答する場合

チェック欄に✓をつけてください。

令和7年度

新潟大学工学部第3年次編入学

学 力 試 験

試 験 科 目	専門基礎科目（プログラミング）	全2頁 (表紙を除く)
---------	-----------------	----------------

注意事項

1. この表紙を含め、全ての試験用紙左上の所定欄に受験番号を記入してください。
2. 解答はその問題と同一の試験用紙に記入してください。解答スペースが足りない場合は、「(裏面に続く)」と明記したうえで、その用紙の裏に続けて解答してください。
3. 各プログラムで解答する科目は以下の表の通りです。科目の選択があるプログラムは表をよく確認の上、科目の過不足がないように注意してください。
4. 解答を行う試験科目には表紙左上のチェック欄に✓を付けてください。✓がない答案は採点されません。
5. 選択しなかった科目についても、表紙に受験番号を記入してください。

学位プログラム	学力試験科目（専門基礎科目）
機械システム工学プログラム	「数学、物理」の2科目
社会基盤工学プログラム	「数学、物理」の2科目
電子情報通信プログラム	「数学、電気回路」の2科目
知能情報システムプログラム	「数学、プログラミング」の2科目
化学システム工学プログラム 応用化学コース	「化学（〔Ⅰ〕有機化学，〔Ⅱ〕無機化学，〔Ⅲ〕物理化学）」
化学システム工学プログラム 化学工学コース	「化学（〔Ⅱ〕無機化学，〔Ⅲ〕物理化学，〔Ⅳ〕化学工学）」
材料科学プログラム	「化学（〔Ⅰ〕有機化学，〔Ⅱ〕無機化学，〔Ⅲ〕物理化学）」 もしくは「数学、物理」の2科目
建築学プログラム	「数学、物理」の2科目
人間支援感性科学プログラム	「数学」（必須）および「物理、電気回路、プログラミング」から1科目 の合計2科目
協創経営プログラム	

受験番号	
------	--

令和7年度
新潟大学工学部第3年次編入学
学 力 試 験

解答は各問とも必ずこの試験用紙に記入すること

試 験 科 目	専門基礎科目 (プログラミング)	1 / 2 頁
---------	------------------	---------

〔I〕以下の問に答えよ。解答は各問の下に記入すること。

- (1) m 個 ($m \geq 1$) の互いに区別のできないボールを, n 個 ($n \geq 1$) の互いに区別のできない袋すべてに詰める仕方の数を $p(m,n)$ と表す。例えば, 5個のボールを3つの袋に詰めるには, $(3,1,1)$, $(2,2,1)$ の2通りなので, $p(5,3) = 2$ である。どんな数 m に対しても, $p(m,1) = 1$, $p(m,m) = 1$ は自明である。袋すべてに詰めねばならないので, $m < n$ に対しては, $p(m,n) = 0$ である。さて, $m > n$ に対して,

$$p(m,n) = p(m-n,n) + p(m-n,n-1) + \cdots + p(m-n,2) + p(m-n,1)$$

が与えられることが分かった。このとき, 以下の問に答えよ。

- ①関数 $p(m,n)$ が既に作成されているとき, キーボードから m と n の値を入力して, $p(m,n)$ の値を出力する main 関数を C 言語で書け。使用した変数やプログラムの動作に関する部分には可能な限りコメントをつけること。

- ② $p(m,n)$ を計算する関数を C 言語で書け。ただし, 再帰呼び出しを用いること。使用した変数やプログラムの動作に関する部分には可能な限りコメントをつけること。

受験番号	
------	--

令和7年度
新潟大学工学部第3年次編入学
学 力 試 験

解答は各問とも必ずこの試験用紙に記入すること

試 験 科 目	専門基礎科目 (プログラミング)	2 / 2 頁
---------	------------------	---------

〔Ⅱ〕次のプログラムは認識した英数字列を順番にリストに格納し、最後に出力するものである。このプログラムを参考にして、あらたに同一の英数字列の出現数をカウントして英数字列とその出現数を出力するプログラムを作成したい。このとき英数字列の出現順序は考慮しない。

- (1) このプログラムのどの部分をどのように変更すれば実現できるか。できる限り効率よく実現できる方法を説明せよ。
- (2) (1) で説明した方法を実現するコードを示せ。変更部分のみを示すだけで良い。このとき英数字列の辞書式順序での大小比較を行う関数 strcmp は使って良い。

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <string.h>
4
5 typedef struct worddata {
6     struct worddata *next;
7     char *str;
8 } worddata;
9 worddata *wordList = NULL;
10
11 int printList()
12 {
13     worddata *car = wordList;
14     while (car != NULL) {
15         printf("%s\n", car->str);
16         car = car->next;
17     }
18     return 0;
19 }
20
21 int main()
22 {
23     int c;
24     char buf[256];
25     int i = 0;
26     while ((c = getchar()) != EOF) {
27         if ( ('a' <= c && c <= 'z') ||
28             ('A' <= c && c <= 'Z') ||
29             ('0' <= c && c <= '9') ) {
30             buf[i++] = c;
31         } else if (i > 0) {
32             worddata *newelem;
33             newelem = malloc(sizeof(worddata));
34             newelem->next = NULL;
35             newelem->str = malloc(i+1);
36             strncpy(newelem->str, buf, i);
37             if (wordList==NULL) wordList = newelem;
38             else {
39                 worddata *car = wordList;
40                 while (car->next != NULL)
41                     car = car->next;
42                 car->next = newelem;
43             }
44             i = 0;
45         }
46     }
47     printList();
48 }
```