

京都産業大学 コンピュータ理工学部
AO 入試 〈情報科目試験型〉

2017 年度入試問題

(2016 年 7 月 xx 日版)

〔I〕 以下の間に答えなさい。

8種類ある記号 A、B、C、D、E、F、G、H を元にした記号列データを考える。

設問 (A)

この A~H で表現された記号のそれぞれを 1 バイト文字としてコンピュータ上で表現した場合、次の 1) 2) 3) の記号列はそれぞれ何バイトとなるか答えよ。

- 1) GEE
- 2) FEAD
- 3) GHGHBC

設問 (B)

この 8 種類の記号に対して、0 と 1 のビットによる表現を行って、それぞれ 3 ビットで表現すること (符号化) を考える。例えば次の表に示すよう符号化した場合、設問 (A) の記号列 1) 2) 3) それぞれはどう表現されるか答えよ。

記号	3 ビット表現による符号
A	0 0 0
B	0 0 1
C	0 1 0
D	0 1 1
E	1 0 0
F	1 0 1
G	1 1 0
H	1 1 1

設問 (C)

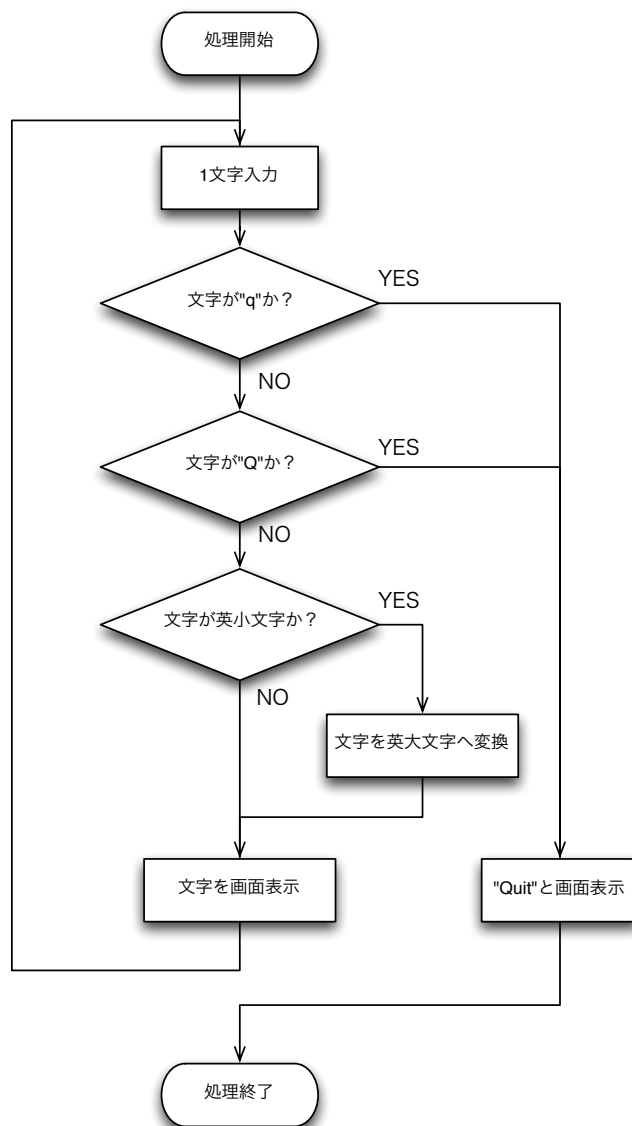
設問 (B) で符号化した各ビット列は、設問 (A) で回答したバイト数から換算すれば少ないビット数で表現できている。1) 2) 3) それぞれ何ビットずつ削減できたか答えよ。

〔Ⅱ〕 以下の問に答えなさい。

次のフローチャートで表されたプログラムに対し、

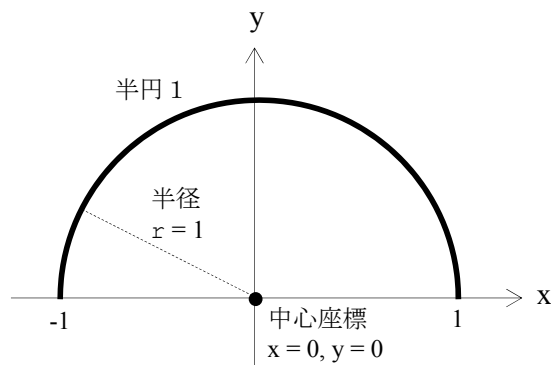
Dragon Quest

と文字入力された場合に画面表示される文字列を答えよ。



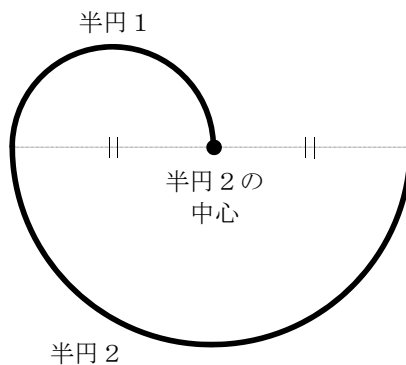
〔III〕 以下の文章を読んで、設問（A）～ 設問（C）に答えなさい。

xy 平面上に上向きあるいは下向きの半円を描く命令 $\text{arc}(x, r, d)$ がある。x は半円の中心の x 座標を表し、r は半円の半径を表す。d は半円の向き（direction）を表す変数で、1 か 2 のどちらかの整数値をとる。d=1 のとき上向き、d=2 のとき下向きの半円を描く。半円の中心の y 座標は常に $y=0$ である。例えば、 $\text{arc}(0, 1, 1)$ を実行すると xy 平面上に次のような半円が表示される。これを半円 1 とする。



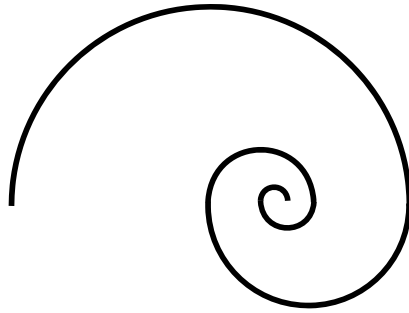
設問（A）

半円 1 を表示した後、半円 1 の右端の点を中心とし、半円 1 の直径の値を半径とするような下向きの半円（これを半円 2 とする）を下図のように表示させた。半円 2 を描く命令を記述せよ。



設問（B）

次に、半円 2 の左端の点を中心として半円 2 の直径の値を半径とする上向きの半円（半円 3）、さらに半円 3 の右端の点を中心として半円 3 の直径の値を半径とする下向きの半円（半円 4）、といった規則で次々と半円を描き加えて、次ページの図のような渦巻き模様を作ることを考える。



半円5まで描き加えたところ

この渦巻き模様を作る[プログラム1]の空欄①～⑤に適切な式または命令文を入れよ。ただし、arc 以外にプログラム中で用いることが出来る命令文と演算子は下の右の枠内に記述した通りとする。また、プログラム中には変数 x, r, d と、繰り返しの回数を表す変数 n 、および定数として整数を用いることができる。

[プログラム1]

```

x ← 0
r ← 1
d ← 1
n ← 1
while n <= 5
  arc( x, r, d )
  if d == 1
    x ← [ ① ]
    [ ② ]
  else
    x ← [ ③ ]
    [ ④ ]
  end
  r ← [ ⑤ ]
  n ← n + 1
end

```

命令文と演算子の説明

<命令文> 変数 ← 式	<動作> 変数に式の値を代入する。
while 条件式 ... end	条件式が真である間、endまでの命令を実行する。
if 条件式 ... else ... end	条件式が真のとき、elseまでの命令を実行し、偽のとき、elseからendまでの命令を実行する。
<比較演算子> A == B A <= B	<意味> A と B の値が等しい。 A の値が B の値以下である。
<四則演算子> + - * /	<意味> 加算 (足し算) を行う。 減算 (引き算) を行う。 乗算 (かけ算) を行う。 除算 (割り算) を行う。ただし、結果は実数で表される。

設問 (C)

設問 (B) のプログラム1は、 d の値に応じた条件分岐 (点線で囲った部分) を、より簡単な計算手続きで同じ結果が得られるように書き換えることが出来る。プログラム1の点線で囲った部分の処理を、条件文 (if ... else ... end) を用いない単純な計算のみで行わせるプログラムコードを示しなさい。必要に応じて、新たな変数 (変数名は独自に考えてよい) を1つだけ用いてよい。

〔IV〕 以下の問に答えなさい。

図 1. に 100 円で三種類の商品が購入できる自動販売機の処理について、簡易な状態遷移図を用いて示す。なおここではモデルを簡素にするために、10 円、100 円しか硬貨の種類が無いものとする。

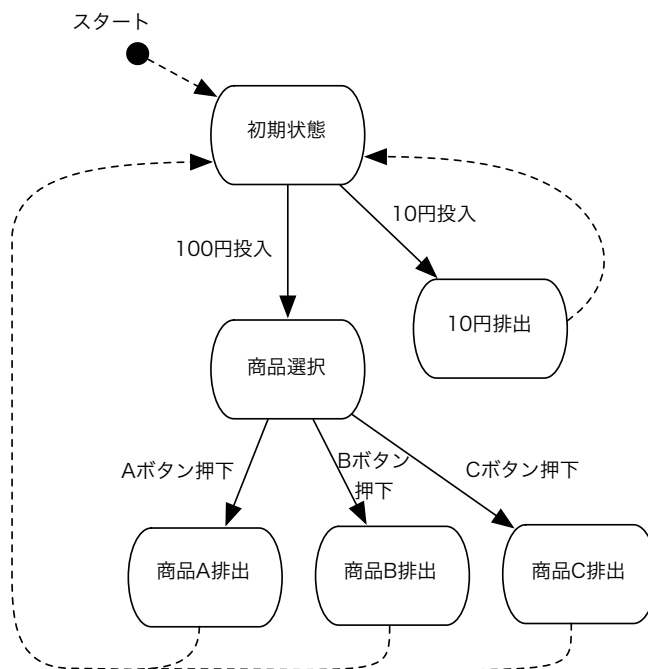


図 1. 100 円で三種類の商品が購入できる自動販売機の状態遷移図

なお、この場で用いる状態遷移図の表記法については図 2. を参照のこと。

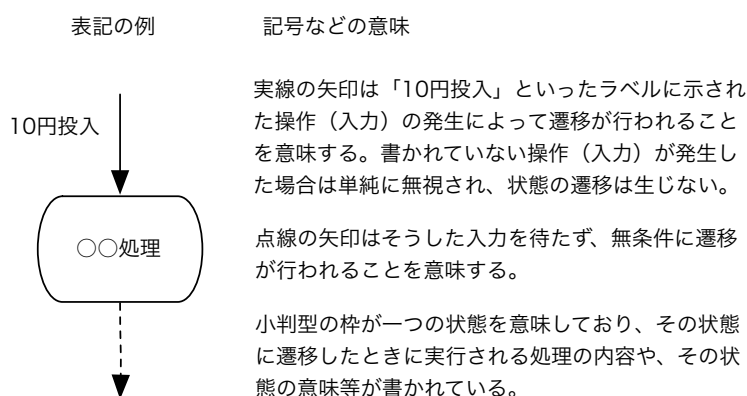


図 2. 図・記号の表記について

つまり、このシステムは以下のように振る舞う。

- このシステムは電源を投入すると、スタートから「初期状態」へと遷移する
- この「初期状態」から、100円が投入されると、「商品選択」状態へと遷移する
- この「商品選択」状態から、どのボタンを押したかによって「商品A排出」「商品B排出」「商品C排出」状態へと遷移する
- それらの各状態では、それぞれ該当する商品を商品出口に排出し、無条件に「初期状態」へと遷移する
- なお、「初期状態」で10円が投入された場合は、「10円保持」状態へと遷移し、10円を排出して返金し、初期状態に戻る

ここで商品が値上がりしたため、110円で商品を買うようにシステムを修正した。修正後の自動販売機の状態遷移図を図3.に示す。

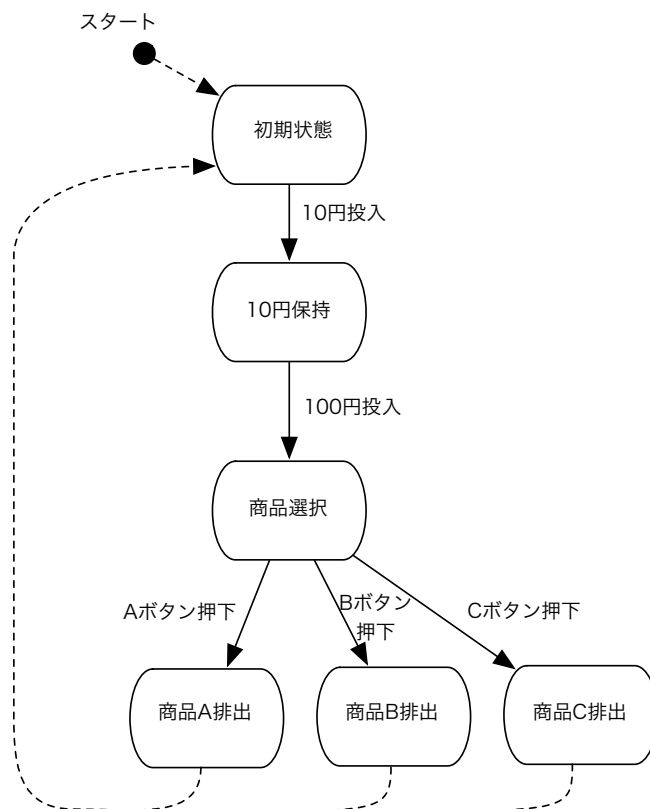


図 3. 110 円での購入に対応した後の状態遷移図

設問 (A)

図 3. に示した自動販売機の動作はしかし不完全である。つまりこの自動販売機は 10 円、100 円の順に硬貨を投入すれば正しく動作するが、100 円、10 円の順に硬貨を入れた場合が想定されていない。そこで 100 円、10 円の順に硬貨を投入しても正しく反応するよう、回答欄の図に修正・加筆せよ。

設問 (B)

ところで図 3. の設計にはまだ考慮漏れがあった。10 円、10 円、と続けて 10 円を投入した場合が考慮されていない。もしそのような事が起きた場合、あとで投入された 10 円を異常入金として返金し、再び「10 円保持」状態に遷移しなければならない。これを実現するよう、回答欄の図に修正・加筆せよ。ただし設問(A)で加えた改良は回答に含まなくて良い。

なお、このようなことを実現するためには、以下のような状態遷移を行う記述を追加する必要がある。

- ・ 10 円の投入によって遷移してくる
- ・ 遷移してくると、10 円を返金口に排出する
- ・ 排出すれば、無条件に次の状態に遷移する

図 4. に、これを実現する記述を例として示す。この問題を含めて、これ以降この記法に従って自分で新しい機能（処理の内容、例えば「10 円排出」など）を定義して使って良い。

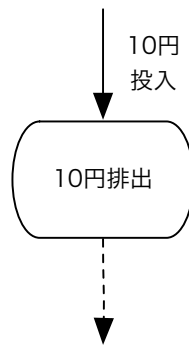


図 4. 10 円を返金する状態

設問 (C)

通常自動販売機には、途中で操作をやめるための「返金レバー」がついている。このレバーを操作した場合、それまでに入金された金額を排出して、初期状態に戻らなければならない。

これを実現し、設問 (A), (B) で改善した部分を含む、完成した自動販売機の状態遷移図を回答欄の図に修正・加筆せよ。

なお返金のための表記は図 5. を参考にして記入されたい。

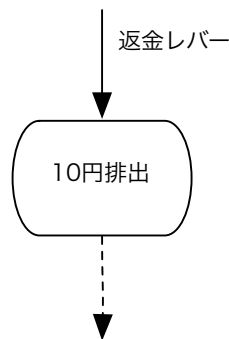


図 5. 返金操作の例

〔V〕 以下の文章を読んで、設問（A）～ 設問（C）に答えなさい。

科学クラブで、伝染性の病気が感染を広げて行く様子をシミュレーションしてみることになり、メンバーがそれぞれにプログラムを作成することになった。

想定する伝染病は、感染してから3～5日の潜伏期間の後に発症し、発症してから2～7日で完治する。他人に感染するのは発症している間だけであり、完治後は同じ病気には感染しない。ある未感染の住人が、ある1日のうちに感染する確率は、(発症者数/都市の人口) × 0.6で与えられるとする。また、シミュレーションは人口10万人の都市で、1日目に発症者が10名いるという状況を想定して行う。

プログラムは、1日目の住民の状態の初期設定を行った後、1日刻みで住民の状態を更新して、病気が感染して行く過程のシミュレーションを行う。

設問（A）

A君は、個々の住民の状態を表すために配列を使うプログラムを考えた。住民の状態として、未感染、発症、完治を挙げたが、友人のBさんからもう一つ状態が必要だという指摘を受けた。それは何だろうか。

設問（B）

A君は最初、要素数10万の整数型の1次元配列を確保し、それぞれの要素が住民一人一人の状態を表すようなプログラムを考えた。しかし、上の設問(A)で考えた数種類の状態を表すだけではシミュレーションができないことに気づいた。さらにどんな情報が必要だろうか。

設問（C）

クラブのメンバーで議論した結果、A君のプログラムはメモリの使用量が必要以上に多いと指摘された。つまり、シミュレーションの結果としては、何日目にどんな状態の住人が何人いるかという情報だけが得られればよい。C君はこの点に注目し、未感染の住民と完治した住民については、配列に個人個人の情報を格納せず、人数だけを記憶しておくプログラムを作成したという。またD君のプログラムでは、確保しなければならない配列は2つで、それぞれの要素数は10個に満たないという。

C君、またはD君の（あるいは別の）メモリ使用量がより少ないプログラムの実現方法の概要を説明しなさい。なお、プログラムコードを書く必要はなく、文章や図、フローチャートなどで説明すればよい。