

令和6年度後期 情報検定

<実施 令和7年2月9日（日）>

基本スキル

(説明時間 13:00~13:10)

(試験時間 13:10~14:10)

- ・試験問題は試験開始の合図があるまで開かないでください。
- ・解答用紙（マークシート）への必要事項の記入は、試験開始の合図と同時に行いますので、それまで伏せておいてください。
- ・試験開始の合図の後、次のページを開いてください。＜受験上の注意＞が記載されています。必ず目を通してから解答を始めてください。
- ・試験問題は、すべてマークシート方式です。正解と思われるものを1つ選び、解答欄の○をHBの黒鉛筆でぬりつぶしてください。2つ以上ぬりつぶすと、不正解になります。
- ・辞書、参考書類の使用および筆記用具の貸し借りは一切禁止です。
- ・電卓の使用が認められます。ただし、下記の機種については使用が認められません。

<使用を認めない電卓>

1. 電池式（太陽電池を含む）以外の電卓
2. 文字表示領域が複数行ある電卓（計算状態表示の一行は含まない）
3. プログラムを組み込む機能がある電卓
4. 電卓が主たる機能ではないもの
 - * パソコン（電子メール専用機等を含む）、携帯電話、スマートフォン、タブレット、電子手帳、電子メモ、電子辞書、翻訳機能付き電卓、音声応答のある電卓、電卓付き腕時計、時計型ウェアラブル端末等
5. その他試験監督者が不適切と認めるもの

＜受験上の注意＞

1. この試験問題は13ページあります。ページ数を確認してください。
乱丁等がある場合は、手をあげて試験監督者に合図してください。
※問題を読みやすくするために空白ページを設けている場合があります。
2. 解答用紙（マークシート）に、受験者氏名・受験番号を記入し、受験番号下欄の数字をぬりつぶしてください。正しく記入されていない場合は、採点されませんので十分注意してください。
3. 試験問題についての質問には、一切答えられません。自分で判断して解答してください。
4. 試験中の筆記用具の貸し借りは一切禁止します。筆記用具が破損等により使用不能となった場合は、手をあげて試験監督者に合図してください。
5. 試験を開始してから30分以内は途中退出できません。30分経過後退出する場合は、もう一度、受験番号・マーク・氏名が記載されているか確認して退出してください。なお、試験終了5分前の合図以降は退出できません。試験問題は各自お持ち帰りください。
6. 試験後の合否結果（合否通知）、および合格者への「合格証・認定証」はすべて、Web認証で行います。
 - ①情報検定（J検）Webサイト合否結果検索ページ及びモバイル合否検索サイト上で、デジタル「合否通知」、デジタル「合格証・認定証」が交付されます。
 - ②団体宛には合否結果一覧ほか、試験結果資料一式を送付します。
 - ③合否等の結果についての電話・手紙等でのお問い合わせには、一切応じられませんので、ご了承ください。

問題 1 次のソフトウェア規模の見積りに関する記述を読み、各設問に答えよ。

ソフトウェア開発では、開発期間や必要な要員数について計画段階から正確に把握することが重要である。ソフトウェア規模の見積りを行う手法の一つにファンクションポイント法がある。

<設問 1> 次のファンクションポイント法に関する記述中の に入れるべき適切な字句を解答群から選べ。

ファンクションポイント法は、ソフトウェア仕様から表に示す 5 項目の分類で機能数を洗い出し、各機能の複雑度に重み付けを行い、下記の式でファンクションポイント(以下 FP とする)を算出する手法である。

各機能の FP 値 = 機能の個数 × 複雑度の重み

ソフトウェアの FP 値 = 各機能の FP 値の合計 × 補正係数

(※)ここでは、補正係数を 0.8 とする。

表に、あるソフトウェアの機能の個数と複雑度の重みおよび、機能の FP 値を示す。

表 機能の個数および複雑度の重み

機能	個数	複雑度の重み	機能の FP 値
外部入力	2	4	8
外部出力	3	6	<input type="text" value="(1)"/>
外部参照	4	8	32
内部論理ファイル	2	5	10
外部インタフェース	3	4	12
機能の合計 FP 値			<input type="text" value="(2)"/>

表において、外部出力の FP 値は であり、すべての機能の FP 値合計は である。したがって、このソフトウェアの FP 値は である。

(1) ~ (3) の解答群

ア. 18

イ. 36

ウ. 48

エ. 64

オ. 72

カ. 80

<設問2> 次の工数計算に関する記述中の□□□□に入れるべき適切な字句を解答群から選べ。

工数は次の式で計算される。

工数[人月] = 人数 × 期間[月]

なお、工数の単位[人月]は一人の開発者が1ヵ月で行う作業量である。

ファンクションポイント法を用いて、新規システム全体の開発工数を見積もったところソフトウェアの合計FP値が600であった。ここで開発者の生産性はすべて同じであり1[人月]あたり15FPとすると、このシステム全体の工数は□□(4)□□[人月]であり、10人で開発を行うように見積もったので、開発期間は□□(5)□□ヵ月である。

当初は10人で開発を行う予定だったが、実際は8人で開発が開始された。この開発者が2人少ない状態で作業が進み、3ヵ月が経過したとき実際に終了した工数は□□(6)□□[人月]であり。このままのペースで作業を続けた場合、残りの工数を終了させるには2ヵ月かかることから開発期間は延びることになる。そこで、計画通り□□(5)□□ヵ月で作業を完了させるために□□(7)□□人増員する必要がある。ただし、増員される開発者も同じ能力とする。

(4) , (5) の解答群

ア. 4

イ. 6

ウ. 32

エ. 40

(6) , (7) の解答群

ア. 8

イ. 12

ウ. 16

エ. 24

問題2 次の浮動小数点数に関する記述を読み、各設問に答えよ。

コンピュータ内部では、小数点以下の値を含む実数値は浮動小数点形式で表現している。IEEE 754における浮動小数点形式では図1のように情報を管理し、数値を $(-1)^{\text{符号}} \times \text{仮数} \times 2^{\text{指数}}$ の形で表現している。

符号部	指数部	仮数部
-----	-----	-----

図1 浮動小数点形式

- ・符号部 (1ビット) 数値が正のとき0、負のとき1を格納する。
- ・指数部 (8ビット) 指数の値に127 を加えたものを格納する。
- ・仮数部 (23ビット) 数値の絶対値を2進数で表したときに、整数の部分に1だけ残すように桁移動 (正規化) した結果の小数部分を左詰めにして格納する。

ここで、10進数の100を上記の浮動小数点形式に変換する。10進数の100は2進数に変換すると $(1100100)_2$ である。これを $(1100100.0)_2 \times 2^0$ ととらえ、整数の部分に1を残すように右へ桁移動して、 $(1.1001)_2 \times 2^6$ とする。これにより、各部の値は次のようになる。

- ・符号部 0
- ・指数部 $6 + 127 = 133 = (10000101)_2$
- ・仮数部 $(1.1001)_2$ の整数の部分を除いた小数部分を左詰めにした値
→ $(1001000\dots)_2$

この値を浮動小数点形式に合わせると、図2のようになる。

0	10000101	100100000000000000000000
---	----------	--------------------------

図2 10進数の100を浮動小数点形式にした結果

<設問1> 次の浮動小数点形式への変換に関する記述中の に入れるべき適切な字句を解答群から選べ。

10進数の-30を浮動小数点形式に変換する。

まず、絶対値の30を2進数にする。

$(30)_{10} = (11110.0)_2 \times 2^0$ であり、整数の部分に1となるように小数点を左へ移動すると、 $(1.1110)_2 \times$ (1) となる。よって、指数部の値は (2) となり、浮動小数点形式に変換したものは (3) である。

(1) の解答群

- ア. 2^2 イ. 2^3 ウ. 2^4 エ. 2^5

(2) の解答群

- ア. 130 イ. 131 ウ. 132 エ. 133

(3) の解答群

- ア.

1	10000001	111100000000000000000000
---	----------	--------------------------
- イ.

1	10000011	111000000000000000000000
---	----------	--------------------------
- ウ.

1	10000100	111100000000000000000000
---	----------	--------------------------
- エ.

1	10000101	111000000000000000000000
---	----------	--------------------------

<設問2> 次の浮動小数点形式から10進数への変換に関する記述中の に入れるべき適切な字句を解答群から選べ。

図3の浮動小数点形式のデータを10進数に変換する。

0	10000001	110000000000000000000000
---	----------	--------------------------

図3 浮動小数点形式のデータ

各部の値を見ると、

- ・符号部は0なので、正の値である。
- ・指数部に表記されている $(10000001)_2$ は (指数+127) の値なので、指数は (4) である。

よって、図3は $(1.11)_2 \times 2^{\text{(4)}}$ を表している。
これを10進数へ変換すると (5) となる。

(4) の解答群

- ア. 1 イ. 2 ウ. 4

(5) の解答群

- ア. 1.75 イ. 3.5 ウ. 7

<設問3> 次の誤差に関する記述に関係の深い字句を解答群から選べ。

- (6) 浮動小数点形式で表す値は仮数部のビット数が決まっているため、変換を途中で止める場合がある。その結果、正確ではない値を格納することになるために誤差が発生する。
- (7) 絶対値の差が極端に大きい2つの数値で加減算を行うと、指数部を合わせるために仮数部を桁移動する。この時、絶対値が小さい方の仮数部が0になってしまう場合に誤差が発生する。
- (8) 絶対値がほぼ等しい数値の間で、同符号の減算や異符号の加算を行うと0に非常に近い結果となる場合がある。これを正規化することで有効桁数が小さくなり、信用できない値が仮数部に含まれることで誤差が発生する。

(6) ~ (8) の解答群

- ア. 打ち切り誤差
- ウ. 桁落ち

- イ. オーバーフロー
- エ. 情報落ち

問題3 次の逆ポーランド記法に関する記述を読み、各設問に答えよ。

数式を表現する方法の一つで、「 $A+B$ 」を「 $AB+$ 」のように2つの演算数の後に演算記号を記述する方式を逆ポーランド記法(後置記法)と呼ぶ。これに対し、「 $A+B$ 」のように普段使っている式は、演算数の間に演算記号がある中置記法と呼ばれる。

中置記法で表した式を2分木で表現すると、容易に逆ポーランド記法に変換できる。

例えば、中置記法の式「 $A-B\times C$ 」の場合、式に演算の優先順位がある場合は、最も低い優先順位の演算子を中心に2分木で表現する。演算記号を親ノード、演算数の子ノードとして式の左から出現する順に左右へ振り分け、図1のように表す。子ノードに演算子が含まれる場合は、そのノードをさらに2分木で表現する。なお、子ノードが演算子を含む式であるものを部分式と表現する。

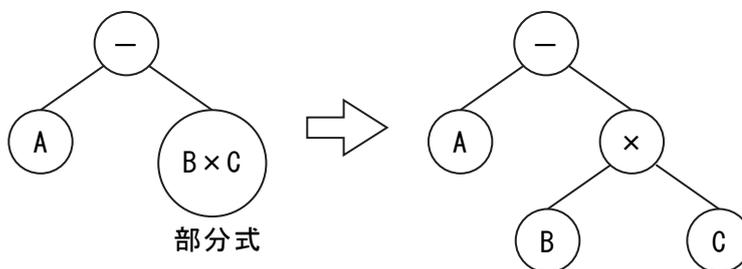


図1 $A-B\times C$ を2分木で表現

2分木から逆ポーランド記法へ変換するには、末端のノードから行う。図1の場合、最初に変換するのは「 $B\times C$ 」の部分であり、左の子ノード、右の子ノード、親ノードの順に走査し「 $BC\times$ 」となり、これを一つのノードとする。そして、二つの子ノードが「 A 」と「 $BC\times$ 」であるものとして「 $ABC\times-$ 」となる。

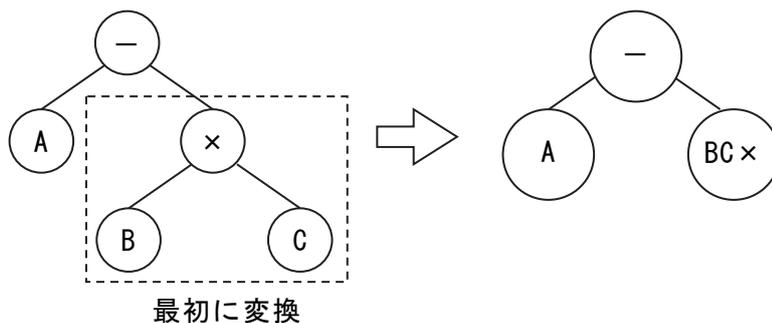


図2 $A-B\times C$ を2分木から逆ポーランド記法へ変換

<設問 1> 次の 2 分木を利用した変換に関する記述中の に入れるべき適切な字句を解答群から選べ。

2 分木を利用して中置記法で表現した式「 $(A+B\div C)\times D$ 」を逆ポーランド記法に変換する。

この式はカッコにより演算の優先順位が変更されているので、演算の優先順位が一番低い演算記号は「 \times 」となり、これを親ノードとする。カッコ内で表現された式は部分式として子ノードにする。式に出現する順に左から子ノードにするので、左子ノードは (1) 、右子ノードは (2) となる。

さらに左子ノードは、部分式内の演算記号である「 $+$ 」を親ノードとし、左子ノードを (3) 、右子ノードは (4) である部分式となる。

さらに部分式内の演算記号である「 \div 」を親ノードとし、左子ノードを (5) 、右子ノードを (6) とした 2 分木を作成する。

完成した 2 分木より逆ポーランド記法で表すと (7) になる。

(1) ~ (6) の解答群

- | | | | |
|--------------|----------|----------------|------|
| ア. A | イ. $A+B$ | ウ. $A+B\div C$ | エ. B |
| オ. $B\div C$ | カ. C | キ. D | |

(7) の解答群

- | | | |
|-----------------------|-----------------------|-----------------------|
| ア. $A+BC\div D\times$ | イ. $AB+C\div D\times$ | ウ. $ABC\div +D\times$ |
|-----------------------|-----------------------|-----------------------|

<設問 2> 次の 2 分木で表現される式に関する記述中の に入れるべき適切な字句を解答群から選べ。

図 3 の 2 分木を式で表現する場合、逆ポーランド記法では (8) となる。

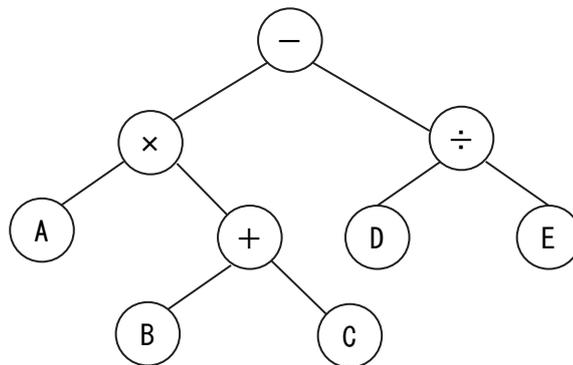


図 3 式を表現した 2 分木

(8) の解答群

- | | | |
|--------------------------|------------------------|-------------------------|
| ア. $A\times B+C-D\div E$ | イ. $ABC+\times-DE\div$ | ウ. $ABC+\times DE\div-$ |
|--------------------------|------------------------|-------------------------|

<設問2> 次の処理時間に関する記述中の に入れるべき適切な字句を解答群から選べ。

3 GHz の CPU を持つ PC で処理を実行するさいの処理時間を求める。1 つの命令を実行するために平均 6 クロックが必要であったとき、1 秒間に実行できる命令数は (6) 命令となり、1 命令の実行時間は平均 (7) 秒となる。この命令を使用して、10,000 命令を実行したときの処理時間は (8) 秒となる。

(6) の解答群

ア. 2×10^8

イ. 3×10^8

ウ. 4×10^8

エ. 5×10^8

(7) , (8) の解答群

ア. 0.2×10^{-8}

イ. 0.4×10^{-8}

ウ. 0.2×10^{-4}

エ. 0.4×10^{-4}

問題5 次の記憶管理に関する記述を読み、各設問に答えよ。

オペレーティングシステムの機能の一つに記憶管理がある。記憶管理の手法に主記憶装置の容量より大きなプログラムの実行を可能にする仮想記憶システムがある。

仮想記憶システムでは、プログラムやデータは補助記憶装置に格納しておき、実行時に必要に応じて主記憶装置に読み込む。こうすることで見かけ上の主記憶装置の容量が増え、大きなプログラムも実行可能となる。このとき、主記憶装置上のメモリを実記憶、補助記憶装置上のメモリ空間を仮想記憶空間と呼ぶ。

<設問1> 次のページング方式に関する記述中の [] に入れるべき適切な字句を解答群から選べ。

ページング方式は、プログラムをページと呼ばれる一定の単位に分割し、このページ単位で転送する方式である。この方式では、実行するページが実記憶のページ枠に存在していない場合、(1) と呼ばれる割込みが発生し、不要なページを実記憶から仮想記憶空間へ追い出し((2))、逆に補助記憶から実記憶に必要なページを読み込む((3))が行われる。なお、(1) が多発すると処理効率が低下する場合があります、これを(4) という。

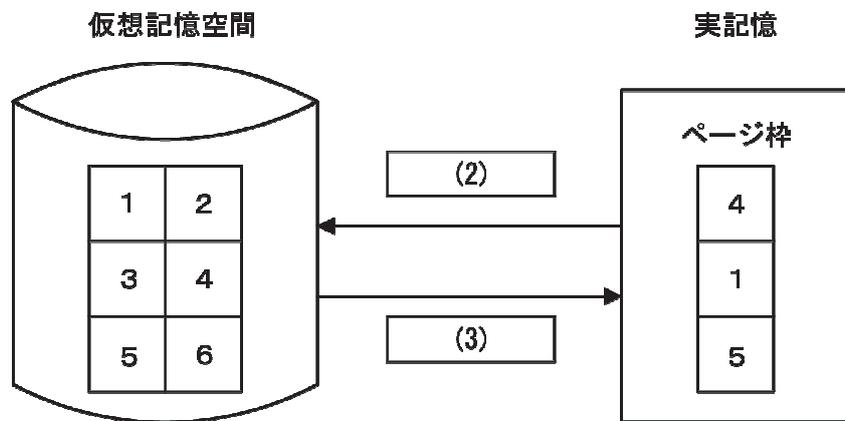


図1 ページング方式

(1) ~ (4) の解答群

- | | |
|------------|------------|
| ア. オーバレイ | イ. スプーリング |
| ウ. スラッシング | エ. セグメント |
| オ. ページアウト | カ. ページイン |
| キ. ページテーブル | ク. ページフォルト |

<設問 2 > 次の仮想記憶システムに関する記述中の に入れるべき適切な字句を解答群から選べ。

設問 1 の図 1 で示されているプログラムでは、仮想記憶空間で 6 ページ分の大きさがあるが、実記憶に割り当てられているページ枠が 3 ページの場合、プログラムで実行したいページは 3 ページ分のページ枠に出し入れをして実行される。

仮想記憶空間のアドレスを仮想アドレスと呼び、図 2 の形式をしている。ページの出し入れのさいには実記憶装置上の物理アドレスに変換して利用することになる。このアドレス変換はオペレーティングシステムによって管理される。

ページ番号	ページ内変位
-------	--------

図 2 仮想アドレスの形式

ページ番号は仮想記憶空間上のページ単位に付けられた番号で、図 1 では 1 ~ 6 である。ページ内変位は各ページの先頭からの相対アドレスであり、1 バイトごとにアドレスが付与される。ここで、1 ページの大きさを $4\text{k}(4096=2^{12})$ バイトとするとページ内変位は最低 (5) ビット必要である。

仮想アドレスから物理アドレスへの変換は、図 3 のようなページテーブルを参照して行われる。添字は仮想アドレスのページ番号を表しており、存在ビットはそのページが実記憶装置に格納されている場合は 1、実記憶装置に格納されていない場合は 0 とする。物理アドレスは実記憶装置に用意されたページ枠の各ページの先頭アドレスを示し、ここでは 10 進数で表記する。図 1 のページ枠の場合のページテーブルは図 3 のようになる。

添字 (ページ番号)	存在ビット	物理アドレス
1	1	9000
2	0	—
3	0	—
4	1	5000
5	1	13000
6	0	—

図 3 ページテーブルの内容

ページテーブルの内容が図 3 の場合、実記憶装置の状態を物理アドレス付きで図 4 に示す。

<設問 3> 次のページリプレースメントに関する記述中の□□□□に入れるべき適切な字句を解答群から選べ。

ページリプレースメントは、実記憶装置と仮想記憶装置間のページの入れ替えのことである。効率をよくするためには、それ以降に参照される確率の低いページを追い出す必要がある。代表的なページリプレースメントアルゴリズムとして次のものがある。

- ・ □□ (7) □□ 方式 …実記憶装置に読み込まれてからの経過時間が最も長いページを追い出す方式。
- ・ □□ (8) □□ 方式 …参照されてからの経過時間が最も長いページを追い出す方式。

(7) , (8) の解答群

ア. FIFO

イ. LFU

ウ. LRU

