

【解答】	
【設問1】	オ
【設問2】	カ
【設問3】	カ
【設問4】	オ
【設問5】	ウ

【解説】製造業における部品の棚卸金額計算に関して、ソフトウェア定義と処理のアルゴリズムを解答するソフトウェア設計の問題である。部品の棚卸しとは、製品を製造するために購入した部品が、使用されずにどのくらい在庫として残っているかを確認する作業である。期間を決め、期間開始時の在庫に期間中の受入数量と払出数量を足し引きし、期間終了時の在庫数と金額を計算する。棚卸しに関する用語を確認しておくとし、受入れとは部品を購入し保管場所に入れること、払出しとは保管場所から取り出すことである。次に、棚卸金額の計算方法だが、問題文の冒頭にもあるように、購入単価は調達時期（購入先を決めたり、購入の契約をしたりする時期）によって変動するため、この問題では、先入先出法を用いている。先入先出法については問題文に説明があるので、そのとおり考えればよい。

【棚卸金額計算の処理】を図示したものが、図1の「IT社が購入した部品の棚卸金額計算処理の流れ」である。これより、受入情報ファイルと払出情報ファイルのソフトウェア定義を考え、それを基に図3の「棚卸計算処理の流れ」を考える。棚卸計算処理では、受入情報ファイルと払出情報ファイルを基に棚卸計算をしているので、ソフトウェア定義を間違えずに行うことが重要であり、ポイントは次の二つである。一つは、棚卸処理で扱うファイルはすべて順ファイルで、受入情報ファイルと払出情報ファイル突き合わせることも、もう一つは、受入情報作成では期首在庫ファイルを入力として使い、棚卸計算では最終的に期末在庫ファイルを作成することである。順ファイル・突合せということから、処理のアルゴリズムはレコードを同じ順番に整列させておき、1件ずつ読み込み比較する事務処理のアルゴリズムはレコードを同じ順番に整列させておくと、期首在庫と期末在庫の関係から棚卸しの期間は期であることがそれぞれ読み取れる。いずれも図1中に記載されており、問題文に記述されている内容の補足事項を図表から読み取る読解力が求められているほか、問題文の記述をロジックとして流れ図と突合わせる論理力も求められている。事務処理のアルゴリズムを学習していない場合は難しく感じただかもしれないが、ソフトウェア定義で使用している項目や先入先出法についても問題文に説明があり、難易度は普通である。

【設問1】順ファイルとして作成された受入情報ファイルのキー項目とその並びを考える。順ファイルとは、順編成ファイルともいい、レコードを先頭から順番に末尾に向かってしか参照できないファイルなので、あらかじめ処理で使用する順番にレコードを整列させておく必要がある。キー項目とはレコードを整列させるための項目、その並びとは設問文の「項目の並びは、左の項目の方が整列の優先度が高い」からも分かるように、キー項目が複数ある場合の整列の優先度である。

まず、キー項目を考える。図2の「受入情報ファイルのレコード様式」を見ると、部品番号、受入日付、購入単価、受入数量が定義されている。部品の棚卸金額計算処理を行うので、当然、部品ごとに金額を計算する必要がある。問題文「棚卸金額計算の処理」(1)に「部品は、部品番号で管理している」とあることから部品番号はキー項目である。受入日付は、「同じ部品を同じ日に複数回受け入れることはない」とあるが、言い換えれば「同じ部品を別の日には受け入れることがある」ということなので、部品番号が重複した場合の整列に必要なキー項目である。

次に、その並びについて考える。図1の「IT社が購入した部品の棚卸金額計算処理の流れ」を見ると、棚卸計算処理には三つの処理があり、受入情報ファイルは受入情報作成処理により作成される出力ファイルであると同時に、棚卸計算処理の入力ファイルでもある。よって、受入情報作成処理では、受入情報ファイルを棚卸計算処理で処理する順番に整列させて作成していることが分かる。これについて、図3の「棚卸計算処理の流れ」の説明には「受入情報ファイルと払出情報ファイルとの突合せによって、～受入情報ファイルの受入日付が古いレコードから順に引き当ていく」とある。これにより、受入情報ファイルは部品番号で整列させ、更に部品番号が同じ場合は受入日付順に整列させておけばよい。したがって、(オ)が正解である。

【設問2】払出情報ファイルに最低限必要な項目を考える。【棚卸金額計算の処理】(2)には「期中払出しファイルを用いて払出数量の集計を行い、払出情報ファイルを出力する」とある。期中とは期首から期末の間を意味するので、期中払出しファイルの内容は、その期の開始日から終了日の間に何をいつ何個使ったかの履歴である。棚卸しはある時点で在庫が何個あるかを確認する作業なので、購入単価が変動する受入情報と違い、払出情報としては部品番号と合計何個使ったかが分かる払出数量があればよいが、期中払出しファイルは期全体の履歴をもっているなので、集計する期間によっては日付の情報が必要になる。しかし、問題文のどこにもある時点がないのかは提示されていない。そこで、図1の「IT社が購入した部品の棚卸金額計算処理の流れ」を見ると、最上部の入力ファイルは期首在庫ファイル、最下部の出力ファイルは期末在庫ファイルなので、部品の棚卸金額計算の期間は期であり、最終的に求めたいのは期末の在庫であると分かる。よって、払出情報ファイルに日付は必要なく、何の部品かを示す部品番号と、その部品をその期に何個払い出したかの払出数量（合計）があればよい。したがって、(カ)が正解である。

【設問3】払出情報ファイルは順ファイルなので、ソフトウェアを作成する手順は、集計処理と集計結果のファイル書き込みの繰返しである。レコードは、ファイルの先頭から順に書き込まれるため、決められた順番に整列させるには、集計処理をその順番で実行する必要がある。よって、集計処理に最低限必要なキー項目として、出力ファイルとなる払出情報ファイルを整列させるためのキー項目を考えればよい。

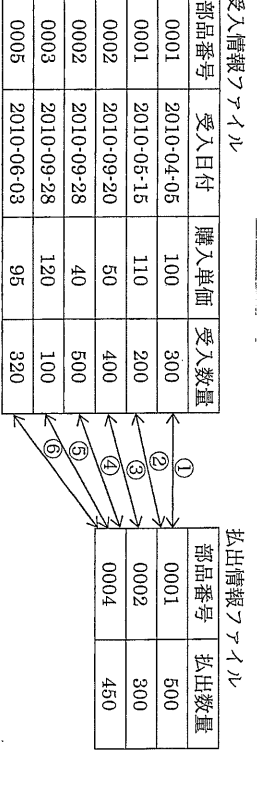
設問2では、払出情報ファイルに最低限必要な項目として部品番号と払出数量を定義したので、どちらの項目で整列させるかを考える。数量で整列させても意味はなく、また、払出情報ファイルは棚卸計算の入力ファイルでもあり、受入情報ファイルと突

き合わせるには、受入情報ファイルと同じ順番に整列されている必要がある。受入情報ファイルのキー項目とその並びは、設問1より部品番号の受入日付順である。よって、払出情報ファイルを整列させるキー項目も部品番号である。したがって、(カ)が正解である。

【設問4】設問1でも確認したが、図3の棚卸計算は、受入情報ファイルと払出情報ファイルとを突き合わせ、払出情報ファイルのレコードの払出数量分を受入情報ファイルの受入日付が古いレコードから順に引き当てる処理である。全部引き当てられた場合は在庫がないが、引き当てられなかった場合は在庫が残り、これを期末在庫ファイルに書き込み、棚卸金額を計算して棚卸金額ファイルに書き込む。突合せとは、事務処理のアルゴリズムの定番の一つで、複数のファイルから1件ずつレコードを読み込み、キー項目の値を比較して同じかどうかで処理内容を振り分けることである。キー項目の値を比較するため、ファイルは同じ順番に整列していることが前提となる。この問題では、受入情報ファイルと払出情報ファイルから1件ずつ読み込み、部品番号を比較する処理が該当する。部品番号が一致すれば引き当てを行い、一致しなかった場合は次のレコードを読み込むか、エラー処理を行う。

図3の「棚卸計算処理の流れ」は、三つの流れ図で構成されている。変数の意味は、受入情報読み込みの流れ図から受入情報ファイルの受入数量がU、突合せキーの部品番号の値がKU、払出情報読み込みの流れ図から払出情報ファイルの払出数量がH、突合せキーの部品番号の値がKHである。

それを踏まえ、流れ図を検証する。設問1～3で作成したソフトウェア定義を基に具体的なレコード内容を作成してケースごと分かりやすい。また、流れ図を検証する過程で設問5についても解答を導くことができる。



KU : KH	H←H-U	H : 0	該当する処理
① 0001=0001	200←500-300 (まだ引当必要)	200>0	(省略)へ分岐、受入情報読み込み
② 0001=0001	0←200-200 (在庫0、期末在庫出力なし)	0=0	C2へ分岐、(P3)、払出情報読み込み
③ 0002=0002	-100←300-400 (引当てより受入れが多い)	-100<0 (在庫あり) (P3)、払出情報読み込み	C1へ分岐、0002の期末在庫書き込み、(P2)、C2に合流、(P3)、払出情報読み込み
④ 0002<0004	— (これ以上、0002の引当てはない)	—	<へ分岐、0002の期末在庫書き込み、金額集計、(P1)
⑤ 0003<0004	— (0003の引当てはない)	—	<へ分岐、0003の期末在庫書き込み、金額集計、(P1)
⑥ 0005>0004	— (0004の受入れがない)	—	>へ分岐、誤り処理、KU、KHに最大値を代入、引当処理終了

図 テストケースのデータと検証結果

このテストデータ（期を2010-04-01～2010-09-30）を基に、棚卸計算処理の流れを見る。

①のケースは、受入情報ファイルと払出情報ファイルが同じ部品番号で引当てをして、払出数量分(H)から受入数量(U)を引いた値が0より小さい場合である。この場合は、まだ払出数量分の引当てができるので、次のレコードの受入情報ファイルを読み込むだけでよい。よって、(省略)へ分岐して、受入情報読み込み処理を行う。

②のケースは、受入情報ファイルと払出情報ファイルが同じ部品番号で引当てをして、受入数量(U)を引いて払出数量分の値(H)がちょうど0となった場合である。問題文に「数量が0となった受入情報ファイルのレコードは、期末在庫ファイルには出力しない」と記述している。これより、その部品に関する期末在庫ファイルへの出力はせず、受入情報ファイルと払出情報ファイルとも次の部品のレコードを読み込んで、新たに突合せ、引当てを行えばよい。C2へ分岐する経路は、P3処理、払出情報読み込み処理を行っているが、これに該当する。よって、P3処理は「受入情報読み込み」が該当する。

③のケースは、受入情報ファイルと払出情報ファイルが同じ部品番号で引当てをして、払出数量分(H)から受入数量(U)を引いた値が0より小さい場合である。払出数量分より受入数量が多いことになるので、在庫が存在することになり、期末在庫ファイルにレコードを書き込む必要がある。よって、C1へ分岐して、期末在庫書き込み処理を行うことになる。先入先出法において、在庫がある場合の棚卸金額は、より後の方に受け入れた部品の金額の合計となるが、受入れのたびに購入単価が違ってくるので、金額の集計を行わなければならない。したがって、P2処理は「金額集計」が該当する。その後、受入情報ファイルと払出情報ファイルの次のレコードを読み込めばよい。

④のケースは、受入情報ファイルの部品番号が払出情報ファイルの部品番号より小さい場合である。＜へ分岐して、期末在庫書き込み処理、金額集計処理、P1処理を行う。③のケースで期末在庫ファイルにレコードを書き込んで、金額集計処理を行った後に、部品を購入することは、それだけ支払いが増えることもある。通常、在庫があればそれ以上に購入することはないが、このケースは、その部品に関して更に過剰受入れをしてしまった場合の処理である。先入先出法において、在庫となる部品の受入れのたびに購入単価が異なるので、プログラム設計としてはそれを考慮してその都度、期末在庫ファイルにレコードを書き込んで、金額の集計を行わなければならない。その後は、受入情報ファイルの次のレコードを読み込めばよい。よって、P1は「受入情報読み込み」となる。

⑤のケースは、④のケースと同じで＜へ分岐して、期末在庫書き込み処理、金額集計処理、P1（受入情報読み込み）処理を行う。この場合は、その部品番号のレコードが、受入払出情報ファイルには存在するが、払出情報ファイルには存在しない場合である。

【解答】	
【設問1】	オ
【設問2】	カ
【設問3】	カ
【設問4】	オ
【設問5】	ウ

【解説】製造業における部品の棚卸金額計算に関して、ソフトウェア定義と処理のアルゴリズムを解答するソフトウェア設計の問題である。部品の棚卸しとは、製品を製造するために購入した部品が、使用されずにどのくらい在庫として残っているかを確認する作業である。期間を決め、期間開始時の在庫に期間中の受入数量と払出数量を足し引きし、期間終了時の在庫数と金額を計算する。棚卸しに関する用語を確認しておくとし、受入れとは部品を購入し保管場所に入れること、払出しとは保管場所から取り出すことである。次に、棚卸金額の計算方法だが、問題文の冒頭にもあるように、購入単価は調達時期（購入先を決めたり、購入の契約をしたりする時期）によって変動するため、この問題では、先入先出法を用いている。先入先出法については問題文に説明があるので、そのとおり考えればよい。

【棚卸金額計算の処理】を図示したものが、図1の「T社が購入した部品の棚卸金額計算処理の流れ」である。これより、受入情報ファイルと払出情報ファイルのソフトウェア定義を考え、それを基に図3の「棚卸計算処理の流れ」を考える。棚卸計算処理では、受入情報ファイルと払出情報ファイルを基に棚卸計算をしているので、ソフトウェア定義を間違えずに行うことが重要であり、ポイントは次の二つである。一つは、棚卸処理で扱うファイルはすべて順ファイルで、受入情報ファイルと払出情報ファイル突き合わせることも、もう一つは、受入情報作成では期首在庫ファイルを入力として使い、棚卸計算では最終的に期末在庫ファイルを作成することである。順ファイル・突合せということから、処理のアルゴリズムはレコードを同じ順番に整列させておき、1件ずつ読み込み比較する事務処理のアルゴリズムはレコードを同じ順番に整列させておくとし、期首在庫と期末在庫の関係から棚卸しの期間は期であることがそれぞれ読み取れる。いずれも図1中に記載されており、問題文に記述されている内容の補足事項を図表から読み取る読解力が求められているほか、問題文の記述をロジックとして流れ図と突合わせる論理力も求められている。事務処理のアルゴリズムを学習していない場合は難しく感じただかもしれないが、ソフトウェア定義で使用している項目や先入先出法についても問題文に説明があり、難易度は普通である。

【設問1】順ファイルとして作成された受入情報ファイルのキー項目とその並びを考える。順ファイルとは、順編成ファイルともいい、レコードを先頭から順番に末尾に向かってしか参照できないファイルなので、あらかじめ処理で使用する順番にレコードを整列させておく必要がある。キー項目とはレコードを整列させるための項目、その並びとは設問文の「項目の並びは、左の項目の方が整列の優先度が高い」からも分かるように、キー項目が複数ある場合の整列の優先度である。

まず、キー項目を考える。図2の「受入情報ファイルのレコード様式」を見ると、部品番号、受入日付、購入単価、受入数量が定義されている。部品の棚卸金額計算処理を行うので、当然、部品ごとに金額を計算する必要がある。問題文「棚卸金額計算の処理」(1)に「部品は、部品番号で管理している」とあることから部品番号はキー項目である。受入日付は、「同じ部品を同じ日に複数回受け入れることはない」とあるが、言い換えれば「同じ部品を別の日には受け入れることがある」ということなので、部品番号が重複した場合の整列に必要なキー項目である。

次に、その並びについて考える。図1の「T社が購入した部品の棚卸金額計算処理の流れ」を見ると、棚卸計算処理には三つの処理があり、受入情報ファイルは受入情報作成処理により作成される出力ファイルであると同時に、棚卸計算処理の入力ファイルでもある。よって、受入情報作成処理では、受入情報ファイルを棚卸計算処理で処理する順番に整列させて作成していることが分かる。これについて、図3の「棚卸計算処理の流れ」の説明には「受入情報ファイルと払出情報ファイルとの突合せによって、～受入情報ファイルの受入日付が古いレコードから順に引き当ていく」とあって、これにより、受入情報ファイルは部品番号で整列させ、更に部品番号が同じ場合は受入日付順に整列させておけばよい。したがって、(オ)が正解である。

【設問2】払出情報ファイルに最低限必要な項目を考える。【棚卸金額計算の処理】(2)には「期中払出しファイルを用いて払出数量の集計を行い、払出情報ファイルを出力する」とある。期中とは期首から期末の間を意味するので、期中払出しファイルの内容は、その期の開始日から終了日の間に何をいつ何個使ったかの履歴である。棚卸しはある時点で在庫が何個あるかを確認する作業なので、購入単価が変動する受入情報と違い、払出情報としては部品番号と合計何個使ったかが分かる払出数量があればよいが、期中払出しファイルは期全体の履歴をもっているなので、集計する期間によっては日付の情報が必要になる。しかし、問題文のどこにもある時点がないのかは提示されていない。そこで、図1の「T社が購入した部品の棚卸金額計算処理の流れ」を見ると、最上部の入力ファイルは期首在庫ファイル、最下部の出力ファイルは期末在庫ファイルなので、部品の棚卸金額計算の期間は期であり、最終的に求めたいのは期末の在庫であると分かる。よって、払出情報ファイルに日付は必要なく、何の部品かを示す部品番号と、その部品をその期に何個払い出したかの払出数量（合計）があればよい。したがって、(カ)が正解である。

【設問3】払出情報ファイルは順ファイルなので、ソフトウェアを作成する手順は、集計処理と集計結果のファイル書き込みの繰返しである。レコードは、ファイルの先頭から順に書き込まれるため、決められた順番に整列させるには、集計処理をその順番で実行する必要がある。よって、集計処理に最低限必要なキー項目として、出力ファイルとなる払出情報ファイルを整列させるためのキー項目を考えればよい。

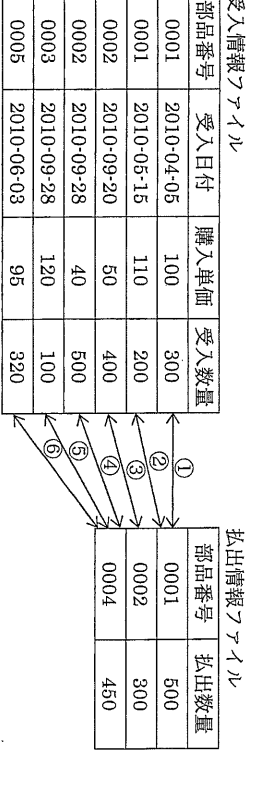
設問2では、払出情報ファイルに最低限必要な項目として部品番号と払出数量を定義したので、どちらの項目で整列させるかを考える。数量で整列させても意味はなく、また、払出情報ファイルは棚卸計算の入力ファイルでもあり、受入情報ファイルと突

き合わせるには、受入情報ファイルと同じ順番に整列されている必要がある。受入情報ファイルのキー項目とその並びは、設問1より部品番号の受入日付順である。よって、払出情報ファイルを整列させるキー項目も部品番号である。したがって、(カ)が正解である。

【設問4】設問1でも確認したが、図3の棚卸計算は、受入情報ファイルと払出情報ファイルと突き合わせ、払出情報ファイルのレコードの払出数量分を受入情報ファイルの受入日付が古いレコードから順に引き当てる処理である。全部引き当てられた場合は在庫がないが、引き当てられなかった場合は在庫が残り、これを期末在庫ファイルに書き込み、棚卸金額を計算して棚卸金額ファイルに書き込む。突合せとは、事務処理のアルゴリズムの定番の一つで、複数のファイルから1件ずつレコードを読み込み、キー項目の値を比較して同じかどうかで処理内容を振り分けることである。キー項目の値を比較するため、ファイルは同じ順番に整列していることが前提となる。この問題では、受入情報ファイルと払出情報ファイルから1件ずつ読み込み、部品番号を比較する処理が該当する。部品番号が一致すれば引き当てを行い、一致しなかった場合は次のレコードを読み込むか、エラー処理を行う。

図3の「棚卸計算処理の流れ」は、三つの流れ図で構成されている。変数の意味は、受入情報読み込みの流れ図から受入情報ファイルの受入数量がU、突合せキーの部品番号の値がKU、払出情報読み込みの流れ図から払出情報ファイルの払出数量がH、突合せキーの部品番号の値がKHである。

それを踏まえ、流れ図を検証する。設問1～3で作成したソフトウェア定義を基に具体的なレコード内容を作成してケースごと分かりやすい。また、流れ図を検証する過程で設問5についても解答を導くことができる。



KU : KH	H←H-U	H : 0	該当する処理
① 0001=0001	200<500-300 (まだ引当必要)	200>0	(省略)へ分岐、受入情報読み込み
② 0001=0001	0←200-200 (在庫0、期末在庫出力なし)	0=0	C2へ分岐、(P3)、払出情報読み込み
③ 0002=0002	-100←300-400 (引当てより受入れが多い)	-100<0 (在庫あり) (P3)、払出情報読み込み	C1へ分岐、0002の期末在庫書き込み、(P2)、C2に合流、(P3)、払出情報読み込み
④ 0002<0004	— (これ以上、0002の引当てはない)	—	<へ分岐、0002の期末在庫書き込み、金額集計、(P1)
⑤ 0003<0004	(0003の引当てはない)	—	<へ分岐、0003の期末在庫書き込み、金額集計、(P1)
⑥ 0005>0004	— (0004の受入がない)	—	>へ分岐、誤り処理、KU、KHに最大値を代入、引当処理終了

図 テストケースのデータと検証結果

このテストデータ（期を 2010-04-01～2010-09-30）を基に、棚卸計算処理の流れを見る。

①のケースは、受入情報ファイルと払出情報ファイルが同じ部品番号で引当てをし、払出数量分 (H) から受入数量 (U) を引いた値が 0 より小さい場合である。この場合は、まだ払出数量分の引当てができるので、次のレコードの受入情報ファイルを読み込むだけでよい。よって、(省略) へ分岐して、受入情報読み込み処理を行う。

②のケースは、受入情報ファイルと払出情報ファイルが同じ部品番号で引当てをし、受入数量 (U) を引いて払出数量分の値 (H) がちやうど 0 となった場合である。問題文に「数量が 0 となった受入情報ファイルのレコードは、期末在庫ファイルには出力しない」と記述している。これより、その部品に関する期末在庫ファイルへの出力はせず、受入情報ファイルと払出情報ファイルとも次の部品のレコードを読み込んで、新たに突合せ、引当てを行えばよい。C2 へ分岐する経路は、P3 処理、払出情報読み込み処理を行っているが、これに該当する。よって、P3 処理は「受入情報読み込み」が該当する。

③のケースは、受入情報ファイルと払出情報ファイルが同じ部品番号で引当てをし、払出数量分 (H) から受入数量 (U) を引いた値が 0 より小さい場合である。払出数量分より受入数量が多いことになるので、在庫が存在することになり、期末在庫ファイルにレコードを書き込む必要がある。よって、C1 へ分岐して、期末在庫書き込み処理を行うことになる。先入先出法において、在庫がある場合の棚卸金額は、より後の方に受け入れた部品の金額の合計となるが、受入れのために購入単価が違ってくるので、金額の集計を行わなければならない。したがって、P2 処理は「金額集計」が該当する。その後、受入情報ファイルと払出情報ファイルの次のレコードを読み込めばよい。

④のケースは、受入情報ファイルの部品番号が払出情報ファイルの部品番号より小さい場合である。＜へ分岐して、期末在庫書き込み処理、金額集計処理、P1 処理を行う。③のケースで期末在庫ファイルにレコードを書き込んで、金額集計処理を行った後に、部品を購入することは、それだけ支払いが増えることもある。通常、在庫があればそれ以上に購入することはないが、このケースは、その部品に関して更に過剰受入れをしてしまった場合の処理である。先入先出法において、在庫となる部品の受入れのために購入単価が異なるので、プログラム設計としてはそれを考慮してその都度、期末在庫ファイルにレコードを書き込んで、金額の集計を行わなければならない。その後は、受入情報ファイルの次のレコードを読み込めばよい。よって、P1 は「受入情報読み込み」となる。

⑤のケースは、④のケースと同じで＜へ分岐して、期末在庫書き込み処理、金額集計処理、P1（受入情報読み込み）処理を行う。この場合は、その部品番号のレコードが、受入払出情報ファイルには存在するが、払出情報ファイルには存在しない場合である。