|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Numer zadania | Numer zadania | Oczekiwana odpowiedź | Maksymalna punktacja za część zadania | Maksymalna punktacja za zadanie |
| **1** | 1.1. | Za poprawne podanie wszystkich wartości F(n) – **2 punkty**  Za podanie wartości z jednym błędem – 1 punkt  Przy dwóch (albo większej liczbie) błędach – 0 punktów.  **Prawidłowa odpowiedź:**   |  |  | | --- | --- | | n | F(n) | | 1 | 2 | | 2 | 6 | | 3 | 24 | | 4 | 120 | | 5 | 720 | | 6 | 5040 | | **2** | **5** |
| 1.2. | Za poprawne podanie wszystkich (6) wywołań F(n) – **2 punkty**  Za podanie poprawnego ciągu wywołań, ale bez pierwszego albo ostatniego wywołania, tzn. bez F(11) albo F(1) – 1 punkt.  **Prawidłowa odpowiedź:**  F(11), F(9), F(7), F(5), F(3), F(1). | **2** |
| 1.3. | Za zakreślenie prawidłowej odpowiedzi – **1 punkt**  **Prawidłowa odpowiedź:**  (n+1)! | **1** |
| **2** | 2.1. | Za poprawne uporządkowanie zbioru – **1 punkt**  **Prawidłowa odpowiedź:**  6 << 8 << 20 << 70 << 100 << 1000 << 35 << 15 << 11 << 3 | **1** | **6** |
| 2.2. | Za poprawne uzupełnienie zbioru – **1 punkt**  **Prawidłowa odpowiedź:**  Prawidłowa odpowiedź powinna utworzyć zbiór złożony z sześciu różnych liczb. Jedną z dwóch brakujących liczb musi być 10. Drugą liczbą może być dowolna liczba nieparzysta (oprócz 7 i 27) albo liczba parzysta większa od 10. | **1** |
| 2.3. | Za poprawnie działający algorytm – **4 punkty**, w tym   * za poprawne wartości początkowe zmiennych – **1 punkt**, * za poprawną organizację pętli – **1 punkt**, * za poprawne porównywanie liczb wg przyjętego porządku dla co najmniej jednego przypadku – **1 punkt,** * za poprawne wyznaczanie elementu maksymalnego dla każdego przypadku (niezależnie od poprawności ustalenia wartości początkowej elementu maksymalnego) – **1 punkt.**   **Przykładowa odpowiedź (1):**   * *max* ⭠ *A*[1] * Dla *i* = 2, 3, ..., *n* wykonaj   + *x* ⭠ *A*[*i*]   + Jeśli *max* jest nieparzyste, to wykonaj     - Jeśli *x* jest nieparzyste oraz *x*<*max*, to   *max* ⭠ *x*   * + *Jeśli max jest parzyste, to wykonaj*     - Jeśli *x* jest nieparzyste lub *x*>*max*, to   *max* ⭠ *x*   * Zwróć *max*   **Przykładowa odpowiedź (2):**   * *max* ⭠ A[1] * dla *i* = 2, 3, ..., *n* wykonaj   + jeżeli (*max* mod 2 = 1 oraz *A*[i] mod 2 =1 oraz *A*[*i*] < *max*), to *max* ⭠ *A*[*i*]   + jeżeli (*max* mod 2 = 0 oraz A[i] mod 2 = 1), to *max* ⭠ *A*[*i*]   + jeżeli (*max* mod 2 = 0 oraz A[i] mod 2 = 0 oraz *A*[*i*] > *max*), to *max* ⭠ *A*[*i*]   **Inne poprawne rozwiązania:**  - algorytm oparty na wyodrębnieniu dwóch podciągów: ciąg liczb nieparzystych i ciąg liczb parzystych, a następnie jeśli ciąg liczb nieparzystych jest niepusty, to znalezienie w nim elementu najmniejszego. W przeciwnym razie znalezienie elementu największego w ciągu liczb parzystych.  - algorytm różniący się od powyższego zastosowaniem sortowania do wyszukiwania elementu najmniejszego/największego. | **4** |
| **3** | 3.1. | Za zaznaczenie kompletu poprawnych odpowiedzi – **1 punkt**  **Prawidłowa odpowiedź:**  F F P | **1** | **4** |
| 3.2. | Za zaznaczenie kompletu poprawnych odpowiedzi – **1 punkt**  **Prawidłowa odpowiedź:**  F P P | **1** |
| 3.3. | Za zaznaczenie kompletu poprawnych odpowiedzi – **1 punkt**  **Prawidłowa odpowiedź:**  F P P | **1** |
| 3.4. | Za zaznaczenie kompletu poprawnych odpowiedzi – **1 punkt**  **Prawidłowa odpowiedź:**  P P F |  |