

1. Jaki będzie wynik wykonania zapytania:

?- [X|Y] = [d, d, c, b, b, a], setof(Z^X, member(Z, Y), W). :

- a) W = [a^d]; W = [b^d]; W=[b^d]; W=[c^d]; W=[d^d]; W=[d^d]; No
- b) W = [a^d]; W=[b^d]; W=[c^d]; W=[d^d]; No
- c) W=[a^d, b^d, c^d, d^d,]; No
- d) W=[a^d, b^d, b^d, c^d, d^d, d^d]; No

2. Elementy programu prologowego, które decydują o jego interpretacji proceduralnej to:

- a) Kolejność reguł w programie
- b) Umiejscowienie odcięć w regułach
- c) Kolejność warunków logicznych
- d) Porządek alfabetyczny reguł

**3. Który z poniższych predykatów należy wstawić w miejsce <puste> aby zapytanie ?-main. Dla programu o treści: main:-repeat,p.
p:-adres(X), write(X), nl, <puste>; true. Spowodowało wypisanie na ekranie (bez interaktywnych nawrotów) wszystkich danych adresowych (pochodzących z faktu postaci adres(X)):**

- a) fail
- b) !
- c) repeat
- d) true

4. Nagłówek w klauzuli prologowej w sensie formalnym:

- a) Może zawierać predykat wyłącznie o arności większej od 0
- b) Może zawierać operatory koniunkcji i dysjunkcji
- c) Może zawierać negację predykatu
- d) Może być pusty

5. Identyfikator zmiennej w języku Prolog:

- a) Ma zasięg globalny
- b) Jest definiowany w momencie deklarowania dziedziny wartości zmiennej
- c) Jest lokalne względem zbioru klauzul o tym samym nagłówku
- d) Jest lokalny względem pojedynczej klauzuli

6. Jakie będą wszystkie możliwe wyniki zapytania:

?- conc([_|_], [X|_], [a, A, b, B, c, C]), var(X).:

- a) X=a; No
- b) X=A; X=B; X=C; No
- c) X=a, X=b, X=c, No
- d) X=b, X=c, No

7. Wartości zmiennych w języku Prolog:

- a) Są ustalane w procesie unifikacji
- b) Muszą być zgodne ze zdefiniowanym wcześniej typem wartości
- c) Mają charakter globalny
- d) Są lokalne względem jednej instancji klauzuli

8. Jaki będzie wynik wykonania zapytania $?-[a|[b]]=..K.$:

- a) $K=[\text{'.'}, a, b]$
- b) $K=[\text{'|'}, a, [b]]$
- c) $K=[\text{'|'}, a, b]$
- d) $K=[\text{'.'}, a, [b]]$

9. Jaki będzie wynik zapytania $?-functor(\text{'f(m,o,n)'}, F, N).$

- a) $F=\text{'f'}$ $N=3$
- b) $F=\text{'|'}$ $N=3$
- c) $F=\text{'.'}$ $N=2$
- d) No

10. Predykaty użytkownika mogą być argumentami wywołania następujących metapredykatów systemowych:

- a) Call
- b) Assert
- c) Bagof
- d) Repeat

11. Jaki będzie wynik wykonania zapytania $?-arg(3, [a, b, c, d], K).$:

- a) $K=[c]$
- b) $K=c$
- c) No
- d) $K=[c, d]$

12. Do grupy predykatów dekompozycji termów języka Prolog zaliczamy następujące metapredykaty systemowe:

- a) Arg
- b) Call
- c) Assert
- d) Findall

13. Które z poniższych zapytań zakończą się spełnieniem celu:

- a) $?- X=a, Y=a, X==Y.$
- b) $?- X=Z, Y=Z, X==Y.$
- c) $?- _==_.$
- d) $?- X=f(_), Y=f(_), X==Y.$

14. Do ewaluacji wartości wyrażenia będącego termem złożonym z funktorem arytmetycznym dochodzi kiedy użyjemy operatora:
- a) =
 - b) =<
 - c) ==
 - d) is
15. Mając na uwadze nie determinizm predykatu retract należy wskazać, które z poniższych odpowiedzi dla celu $?- g(X)$. są poprawne, jeżeli wcześniej zostały wydane zapytania $?- \text{assertz}(g(1)), \text{assertz}(g(2)), \text{assertz}(g(3))$.
 $?- \text{retract}(g(X))$. :
- a) $X=1; X=2; X=3$; No
 - b) $X=2; X=3$; No
 - c) $X=3$; No
 - d) No
16. Która z interpretacji deklaracyjnych definicji klauzuli $p:-!, a,b,!,c. p:-d$. jest kompletna i poprawna:
- a) $P \Leftrightarrow (a \wedge b \wedge c) \vee (\sim a \wedge \sim c \wedge d)$
 - b) $P \Leftrightarrow (a \wedge b \wedge c)$
 - c) $P \Leftrightarrow (a \wedge b \wedge c) \vee (\sim a \wedge d) \vee (\sim c \wedge d)$
 - d) $P \Leftrightarrow (\sim a \wedge \sim c) \vee d$
17. Dla następującego programu prologowego: $p(N) :- N=1; N=2, !; N=3, !$. ,które z poniższych odpowiedzi są wszystkimi rozwiązaniami dla zapytania $?- p(X), !, p(Y)$. :
- a) $X=1 Y=1; X=1 Y=2; X=2 Y=1; X=2 Y=2$
 - b) $X=1 Y=1; X=1 Y=2$
 - c) $X=1 Y=1; X=1 Y=2; X=1 Y=3;$
 $X=2 Y=1; X=2 Y=2; X=2 Y=3;$
 $X=3 Y=1; X=3 Y=2; X=3 Y=3$
18. Argumentami/predykatami predykatów (wykluczamy metapredykaty systemowe) mogą być:
- a) Wyłącznie atomy i/lub zmienne
 - b) Wyłącznie termy złożone
 - c) Inne predykaty
 - d) Dowolne termy

19. Weryfikacji rodzaju termu odbywa się w języku Prolog za pomocą następujących predykatów systemowych :

- a) Var
- b) Functor
- c) Setoff
- d) Atomic

20. Dla definicji operatorów `op(100, xfy, #)`. oraz `op(100, fy, @)`. Wyrażenie `a # @ b # c` jest:

- a) Równoważne wyrażeniu `a # (@ b # c)`
- b) Równoważne wyrażeniu `a # ((@ b) # c)`
- c) Niepoprawne
- d) Równoważne wyrażeniu `a # @ (b # c)`