Matematyka dyskretna dla informatyków. Wykład 1

1 Teoria zbiorów

1.1 Pojęcie zbioru

Pojęcie zbioru jest na pozór bardzo intuicyjne. O zbiorach uczą się dzieci już w pierwszej klasie szkoły podstawowej. Jest ono tak zakorzenione w czasie przerabiania całego programu szkolnego matematyki, że nie zauważa się niebezpieczeństw związanych z jego stosowaniem. Po raz pierwszy na kłopoty związane z dowolnością określenia zbioru zwrócił uwagę w połowie XIX wieku Bolzano, a rozwinął badania nad zbiorami wybitny matematyk niemiecki Georg Cantor pod koniec XIX wieku. Jego odkrycia były tak niespodziewane, że wywołały gwałtowną reakcję niektórych jego kolegów, posuwającą się niekiedy do odsądzania od czci i wiary twórcę teorii zbiorów.

Podstawową obserwacją Cantora było to, że nie każdy zbiór, który jesteśmy w stanie ,,określić" istnieje. Niektóre konstrukcje myślowe bowiem prowadzą do sprzeczności i sprawiają, że to, co nam się zdawało poprawnym określeniem zbioru, nie jest w ogóle definicją żadnego obiektu.

Nie ma problemów, dopóki używamy zbiorów skończonych, czyli takich, których elementy można podać przez wyliczenie w skończonym czasie. Przykłady takich zbiorów, to $\{1,3\}$, $\{1\}$, $\{\{1,3\},1\}$, $\{\{1,3\},\{1\}\}\{\{\{1,3\},\{1\}\}\}$. Nazwijmy je kolejno Z_1, Z_2, Z_3, Z_4, Z_5. Pierwszy z tych zbiorów, Z_1 ma 2 elementy: 1 oraz 3. Drugi,Z_2, jest zbiorem jednoelementowym zawierającym element 1, trzeci, Z_3, dwuelementowym; jednym jego elementem jest pierwszy z wyżej zdefiniowanych zbiorów, a drugim liczba 1. Czwarty z tych zbiorów, Z_4 też ma 2 elementy: oba są zbiorami. Piąty , Z_5 jest zbiorem jednoelementowym; jego jedynym elementem jest zbiór Z_4.

Bardzo ważnym zbiorem jest zbiór pusty, nie zawierający żadnego elementu i oznaczany przez $\emptyset$. Przyjmuje się, że wszystkie zbiory puste są sobie równe, czyli inaczej mówiąc, zbiór pusty jest tylko jeden. Oczywiście zbiór $\{\emptyset\}$ zawierający jako jedyny element zbiór pusty, sam pusty nie jest. Przyjmujemy konwencję, według której elementy zbioru powtarzać się nie mogą[1], a kolejność ich wymieniania nie wpływa na zawartość zbioru. Dla uniknięcia nieporozumień przyjmiemy, że jeśli przy wyliczaniu elementów zbioru jakiś element się powtórzy, to takie powtórzenie ignorujemy. Tak więc zapis $\{1,1,2\}$ będzie oznaczał zbiór $\{1,2\}$, który jest identyczny ze zbiorem $\{2,1\}$.

Właściwie najlepiej można określić zbiór, jako coś, co posiada pewne elementy, przy czym dla każdego obiektu powinniśmy być w stanie stwierdzić, czy do zbioru należy, jako jego element, czy nie. Fakt należenia elementu x do zbioru A oznaczamy przez $x \in A$.

Typ obiektów należących do jednego zbioru może być bardzo różny i nawet w ramach jednego zbioru mamy do czynienia czasami z obiektami zupełnie różnych typów. Przykładem takiego zbioru jest zbiór Z_3, którego elementami są jednocześnie liczba oraz zbiór. W informatyce często mamy do czynienia ze zbiorami jednakowego typu i niektóre języki programowania (np. Pascal) umożliwiają bezpośrednio operowanie na obiektach typu zbiorowego. Implementacja zbiorów niejednorodnych co do typu elementów wymaga pewnych specyficznych technik realizacyjnych. Uwaga: do żargonu informatycznego

[1]obiekty matematyczne będące odpowiednikami zbiorów, z tą różnicą, że elementy mogą się powtarzać, nazywane są wielozbiorami, multizbiorami lub zbiorami z powtórzeniami

przeniknęło niestety używanie określenia *zbiór* zamiast *plik*, kiedy mowa jest o zapisach na dysku. Nie jest mi znana geneza tej praktyki — w każdym razie pliki dyskowe nie mają wiele wspólnego ze zbiorami: są one najczęściej uporządkowane, elementy wchodzące w ich skład mogą się powtarzać, nie można na nich wykonywać żadnych standardowych operacji zbiorowych. Słowo *zbiór* i tak jest przeciążone, więc nadużywanie go w jeszcze jednym znaczeniu jest nieporozumieniem.

W końcu XIX wieku matematycy pokusili się o aksjomatyzację teorii zbiorów (nazywanej po polsku też teorią mnogości). Układ aksjomatów zaproponowany na początku XX wieku przez dwóch matematyków: Zermelo i Fraenkla pozwala na uściślenie pojęcia zbioru, ale nie będziemy go tu przytaczali, odwołując się do intuicyjnego pojęcia zbioru i wyjaśniwszy nieporozumienia, jakie mogą płynąć z radosnego podejścia do problemów związanych ze zbiorami.

1.2 Kłopoty ze zbiorami

Co jest złego w przyjęciu, że zbiorem może być wszystko? Każda grupa obiektów, jeżeli tylko sobie tego zażyczymy, powinna dać się zebrać w zbiór. Okazuje się, że takie naiwne podejście zawiera nieoczywiste pułapki. Za chwilę podamy próbę konstrukcji zbioru, która doprowadzi do sprzeczności.

Postawmy sobie pytanie: czy zbiór może być elementem samego siebie. To znaczy, czy może zachodzić $A \in A$ dla pewnego zbioru A? Konstrukcja takiego zbioru w zasadzie powinna być możliwa. Niełatwo sobie wyobrazić, jak taki twór mógłby wyglądać, choć, jak zaraz się przekonamy, coś w rodzaju przykładu takiego obiektu będziemy mogli zobaczyć na własne oczy. Na razie nie wiemy, czy takie zbiory istnieją, ale nam to nie przeszkadza, bo jeśli nawet nie istnieją, to po prostu zbiór złożony ze zbiorów, które są swoimi własnymi elementami jest pusty. Niewątpliwie większość zbiorów powinny stanowić zbiory, które same do siebie nie należą. Nazwijmy je roboczo ,,normalnymi". Utwórzmy zatem zbiór Z złożony z takich właśnie ,,normalnych" zbiorów, które do siebie nie należą. Postawmy teraz pytanie, czy $Z \in Z$? Jeżeli $Z \in Z$, to znaczy, że Z jest ,,nienormalny", a zatem nie należy do siebie, na mocy swej definicji. Czyli z tego, że $Z \in Z$ wynika, że $Z \notin Z$. Sprzeczność. Zatem jedyne, co nam pozostaje, to przyjąć, że $Z \notin Z$. Ale wtedy nasz zbiór Z jest ,,normalny", więc $Z \in Z$. Kolejna sprzeczność.

Nie da się tego cyklu przerwać. Po prostu do sprzeczności przywiodło nas założenie, że istnieje zbiór Z. Tymczasem takiego zbioru po prostu nie ma. Paradoks ten zauważony został po raz pierwszy przez wielkiego matematyka angielskiego, Bertranda Russella i tak naprawdę nigdy do końca nie został rozwiązany. Unika się tego paradoksu przyjmując pewne ograniczenia na sposoby określania zbiorów.

Żeby pozbyć się wrażenia o akademickości tego problemu, podamy przykład, tego, że w informatyce można się zetknąć z paradoksem Russella naocznie. Dla tych, którzy znają system DOS[2], wiadomym będzie, że w każdym katalogu możemy mieć rozmaite podkatalogi oraz pliki. System katalogów normalnie tworzy drzewo opisane w specjalnej tablicy systemowej zwanej FATem (File Allocation Table). Można zatem powiedzieć, że katalog, to jest zbiór jego podkatalogów (być może pusty) oraz zbiór znajdujących się w nim plików (też może być pusty). Warto zauważyć, że podstawowe właściwości zbiorów są

[2]... i nie tylko: większość istniejących systemów operacyjnych jest podatna na opisywaną poniżej aberrację

tu zachowane: kolejność wymienienia elementów nie ma wpływu na zawartość katalogu, jak również niemożliwe jest powtórzenie nazwy jakiegokolwiek obiektu w ramach danego katalogu.

Okazuje sie, że prostą sztuczką hakerską można podkatalogiem danego katalogu uczynić ten sam katalog. Wystarczy w odpowiednim miejscu w tablicy FAT wpisać adres tego katalogu, jako adres jednego z jego podkatalogów. System cokolwiek zgłupieje. To znaczy najczęściej będzie próbował przy wykonywaniu komend przechodzenia do podkatalogu wchodzić po prostu po raz kolejny do tego samego katalogu, w którym będziemy widzieli kolejną instancję katalogu, w którym się cały czas znajdujemy. Program Norton Commander na przykład będzie posłusznie wydłużał ścieżkę dostępu o kolejne wcielenia tego samego katalogu, a wygląd zawartości katalogu nie będzie się zmieniał ani na jotę. Co gorsza, pozbyć się tego dziwoląga wcale nie jest prosto: w szczególności system zabezpiecza się przed usunięciem katalogu, który jest niepusty i coś zawiera. O ile usunięcie wszystkich plików i innych podkatalogów jest łatwe, o tyle usunięcie samego siebie jako podkatalogu powoduje w najlepszym przypadku odmowę z komentarzem, że nie można usuwać niepustych podkatalogów.

Wywołanie w takiej sytuacji programu próbującego rozwinąć podkatalogi, np. wyświetlającego pełne drzewo podkatalogów, najczęściej powoduje zawieszenie się systemu.

Tak oto paradoks Russella z teorii zbiorów ma swoją ilustrację w informatyce. Próba zrobienia katalogu, który jest swoim podkatalogiem doprowadza do poważnych kłopotów. Takich sytuacji w informatyce jest znacznie więcej. W szczególności trzeba być ostrożnym gdy mamy do czynienia ze strukturami definiowanymi rekurencyjnie, o czym będzie mowa w kolejnych rozdziałach, gdyż moment nieuwagi może spowodować zapętlenie się algorytmu tak, jak to miało miejsce w opisywanym przykładzie z katalogami.

Oto inne znane przykłady sprzeczności oparte na paradoksie Russella.

Przykład 1. *Król pewnego niewielkiego państwa wydał edykt, na mocy którego jedyny golibroda w królestwie miał golić tych i tylko tych mieszkańców, którzy się sami nie golą.* Rozważ, czy golibroda powinien sam się golić, czy nie.

Przykład 2. *Niektóre ciągi liter i cyfr opisują w sposób jednoznaczny pewne liczby. Przykładowo ciągi: '1', 'jeden', 'liczba o jeden większa niż zero' opisują liczbę* 1*. Dla odmiany ciąg 'liczba, która podniesiona do kwadratu daje jeden' nie jest takim ciągiem, bo nie wiadomo, czy ma określać* 1*, czy* -1*. Podobnie nie jest takim ciągiem ciąg 'Ala ma kota', ani 'pppqqq'. Rozważmy zbiór wszystkich napisów co najwyżej* 1000*-literowych i wybierzmy z nich te, które określają jednoznacznie pewne liczby. Takich napisów jest skończona liczba, więc zbiór liczb możliwych do określenia za pomocą tych napisów będzie skończony. Skoro tak, to będzie w tym zbiorze istniała liczba największa. Zdefiniujmy liczbę* X*, jako liczbę o 1 od niej większą.* Powyższy tekst definiuje pewną liczbę. Zauważ, że tekst ten jest pewnym napisem o długości nie przekraczającej tysiąca znaków. Czy widzisz sprzeczność?

1.3 Działania i zależności między zbiorami.

Ważną cechą zbiorów jest możliwość wykonywania na nich działań. Najczęściej mamy do czynienia z sytuacją, w której rozważane zbiory są podzbiorami pewnego zbioru złożonego ze wszystkich interesujących nas elementów. Taki zbiór oznaczymy literą U i będziemy nazywali przestrzenią. Czasami, kiedy zbiór U jest oczywisty nie będziemy w ogóle o nim wspominali.

Ważnym narzędziem umożliwiającym definiowanie podzbiorów jest zapis

$$X = \{x \in U : P(x)\},$$

która tworzy zbiór X złożony z tych elementów x zbioru U, które spełniają własność $P(x)$. Na przykład zbiór $\{x \in I\!N :$ x jest liczbą pierwszą parzystą definiuje zbiór jednoelementowy $\{2\}$. Istotne jest, aby używać jej tylko wtedy, gdy jest dany zbiór U. Pozwala to uniknąć paradoksu Russella.

Powiemy, że zbiór A *zawiera się* w zbiorze B (albo, równoważnie, jest *podzbiorem* B), wtedy i tylko wtedy, gdy każdy element zbioru A jest elementem zbioru B. Oznaczamy tę sytuację przez $A \subseteq B$. Z kolei $A = B$ wtedy i tylko wtedy, gdy $A \subseteq B$ oraz $B \subseteq A$. Jeżeli $A \subseteq B$ i $A \neq B$, to mówimy, że A jest podzbiorem *właściwym* zbioru B i oznaczamy to przez $A \subset B$.

Niech $A, B, C \subseteq U$. Rozważamy następujące działania:

$$-A = \{x \in U : x \notin A\}$$

$$A \cup B = \{x \in U : x \in A \text{ lub } x \in B\}$$

$$A \cap B = \{x \in U : x \in A \text{ oraz } x \in B\}$$

$$A \backslash B = \{x \in U : x \in A \text{ oraz } x \notin B\}$$

Działania te są nazywane odpowiednio dopełnieniem zbioru, sumą dwóch zbiorów, ich iloczynem (albo przecięciem) i różnicą. Zauważmy, że $A \setminus B \neq A \cup (-B)$, więc nie ma analogii z dodawaniem i odejmowaniem liczb. Definiuje się też czasami operator *różnicy symetrycznej zbiorów* i ze względu na jego użyteczność wprowadzimy go do naszego repertuaru:

$$A \oplus B = \{x : x \in A \text{ lub } x \in B \text{ oraz } x \notin A \cap B$$

Zatem różnica symetryczna złożona jest z tych elementów zbiorów A i B, które rozróżniają A i B. Bezpośrednio z definicji wynikają równości:

$$A \oplus B = (A \cup B) - (B \cap A) = (A - B) \cup (B - A)$$

Zachodzą następujące równości:

$$(A \cup B) \cup C = A \cup (B \cup C) \quad \text{łączność sumy}$$

$$A \cup A = A \quad \text{idempotentność sumy}$$

$$A \cap A = A \quad \text{idempotentność iloczynu}$$

$$-(-A) = A \quad \text{prawo podwójnego dopełnienia}$$

$$(A \cap B) \cap C = A \cap (B \cap C) \quad \text{łączność iloczynu}$$

$$A \cup B = B \cup A \quad \text{przemienność sumy}$$

$$A \cap B = B \cap A \quad \text{przemienność iloczynu}$$

$$(A \cap B) \cup C = (A \cap C) \cup (B \cap C) \quad \text{rozdzielność sumy względem iloczynu}$$

$$(A \cup B) \cap C = (A \cup C) \cap (B \cup C) \quad \text{rozdzielność iloczynu względem sumy}$$

$$-(A \cup B) = (-A) \cap (-B) \quad \text{prawo negacji sumy}$$

$$-(A \cap B) = (-A) \cup (-B) \quad \text{prawo negacji iloczynu}$$

Dwa ostatnie prawa nazywają się też *prawami de Morgana.*

$$A \cup \emptyset = A$$

$$A \cup U = U$$

$$A \cap \emptyset = \emptyset$$

$$A \cap U = A$$

Cztery ostatnie prawa nazywane są prawami identyczności.

Bardzo ważną konstrukcją jest tworzenie *uporządkowanych par* z elementów dwóch zbiorów. Jeżeli $a \in A$ oraz $b \in B$, to przez (a, b) będziemy rozumieli uporządkowaną parę złożoną z a i z b. Uwaga: elementy a i b nie muszą być różne, a dodatkowo $(a, b) = (b, a)$ wtedy i tylko wtedy gdy $a = b$. Zatem, jak sama nazwa wskazuje, kolejność elementów jest w parze uporządkowanej, istotna. Zbiór $\{(a, b) : a \in A, b \in B\}$ jest nazywany *iloczynem kartezjańskim* zbiorów A oraz B oraz oznaczany przez $A \times B$. W szczególności ma sens pojęcie *kwadratu kartezjańskiego* zbioru A, który określamy jako $A \times A$.

Przykład Niech $A = \{1, \{1\}\}, B = \{1, 2, 3\}$.
Wtedy $A \times B = \{(1, 1), (1, 2), (1, 3), (\{1\}, 1), (\{1\}, 2), (\{1\}, 3)\}$.

Formalnie po raz pierwszy definicję pary uporządkowanej podał wybitny polski matematyk, Kaziemierz Kuratowski, który parę (a, b) zdefiniował jako zbiór $\{\{a, b\}, a\}$. Przy tej definicji podaje się oba elementy, które mają w takiej parze wystąpić oraz wskazuje się na ten, który ma być pierwszy. Warto zwrócić uwagę na to, że w przypadku pary uporządkowanej nie żądamy różności elementów a i b.

Pojęcie iloczynu kartezjańskiego uogólnia się na dowolne skończone ciągi zbiorów $A_1, \ldots, A_n$ przyjmując, że $A_1 \times A_2 \times, \ldots, A_n = \{(a_1, a_2, \ldots, a_n) : a_1 \in A_1, a_2 \in A_2, \ldots, a_n \in A_n\}$.

Zadania

1. Pokaż, że jeśli $a \in \{\{b\}\}$, to $b \in a$

2. W pewnym kraju znajduje się centralna biblioteka, która zarządza bibliotekami lokalnymi. Każda z bibliotek na życzenie dyrektora centralnej biblioteki miała przygotować wydany w formie książkowej katalog wszystkich książek znajdujących się w niej, dołączyć taki katalog do swego księgozbioru, a drugi egzemplarz przesłać do centrali. Niektóre z przysłanych katalogów zawierały odnośniki do siebie samych, inne nie. Dyrektor centralnej biblioteki polecił przygotować wydanie książkowe zawierające listę wszystkich tych katalogów, które są we wszystkich bibliotekach (w tym w centralnej) i które nie zawierają odnośnika do samego siebie. Czy ten katalog powinien zawierać odnośnik do samego siebie?

3. Zadanie 2/34, 4/35, 9/35, 6/35, 7/35, 8/35, 12/36, 15/36,17/36(b,c) z „Matematyki dyskretnej” Rossa i Wrighta.

Te z zadań, których się nie zdąży zrobić na zajęciach należy zrobić w domu. Jest to generalna zasada dotycząca przyszłych scenariuszy.