

Приложение F. Сравнение на UniPascal с ISO Pascal

В това приложение се прави сравнение на UniPascal с ISO Pascal, както е описан в книгата на K.Jensen N.Wirth, Pascal User Manual and Report, Springer-Verlag, 1974, 1985 (ISBN 3-540-96048-1).

F.1. Нереализирани възможности на ISO Pascal

- В ISO Pascal идентификаторът може да бъде с произволна дължина, като всички символи от името са значещи. В UniPascal идентификаторът може да бъде с произволна дължина, но само първите осем символа са значещи.
- В ISO Pascal символът @ е алтернативно представяне на символа ^. В UniPascal символът ^ няма алтернативно представяне.
- В ISO Pascal е разрешено използването на различни коментарни скоби в един коментар, т.е. коментарът може да започва с { и да завършва с } и обратно - да започва с { и да завършва с *}. В UniPascal това не е разрешено. Коментарите трябва да започват и завършват с един и същ вид коментарни скоби.
- В ISO Pascal всяка стойност от множеството стойности, принадлежащи на типа на полето от вариантната част на запис, трябва да се указва. В UniPascal това изискване не се налага.
- В ISO Pascal всяка променлива от файлов тип има асоциирана с нея буферна променлива, отбелязвана с ^ след името на файловата променлива. В UniPascal файловете променливи нямат асоциирана буферна променлива и използването на ^ след името на файлова променлива е грешка.
- В ISO Pascal полето-селектор от вариантната част на запис не може да бъде използвано в качеството на фактически параметър-променлива. В UniPascal това изискване не се налага.
- В ISO Pascal са дефинирани процедури-параметри. В UniPascal процедури-параметри не са реализирани.
- В ISO Pascal стандартните процедури RESET и REWRITE не изискват име на физическия файл. В UniPascal синтаксисът на стандартните процедури RESET и REWRITE е разширен и изисква име на външен файл.
- В ISO Pascal са дефинирани стандартните процедури GET и PUT, които се използват за четене от и писане във файлове. В UniPascal тези процедури не са реализирани.
- В ISO Pascal са дефинирани стандартните процедури PACK и UNPACK, които се използват за 'пакетиране' и 'разпакетиране' на пакетирани променливи. В UniPascal тези процедури не са реализирани.
- В ISO Pascal операцията $I \bmod J$ винаги връща положителен резултат и е грешно да се използва, ако J е 0 или отрицателно. В UniPascal $I \bmod J$ се пресмята като $I - (I \text{ div } J) * J$ и може да се използва при отрицателни стойности на J .
- В ISO Pascal операторът GOTO може да използва дефиниран в по-външен блок етикет. В UniPascal това не е разрешено и предизвиква грешка по време на компилация.
- В ISO Pascal избиращият израз (selector) в CASE оператора трябва да е равен на една от използваните константи. В UniPascal това не се изисква. Ако избиращият израз не е равен

на нито една от тях, изпълняват се операторите в ELSE частта на CASE оператора. Ако такава няма, не се изпълнява нищо.

- В ISO Pascal не се разрешава промяната на управляващата променлива на FOR цикъла. В UniPascal това не се проверява.
- В ISO Pascal стандартната процедура READ присвоява интервал на променлива от тип CHAR, ако преди четенето EOLN = TRUE. В UniPascal на променливата от тип CHAR в тази ситуация се присвоява <cr> (ASCII код 13).
- В ISO Pascal при използване на стандартната процедура WRITE в текстов файл променлива от тип пакетирани низ се отрязва, ако указаната дължина е по-малка от необходимата. В UniPascal винаги се записва цялата променлива, независимо че указаната дължина е недостатъчна.
- В ISO Pascal е разрешено използването на променливи от логически тип в стандартната процедура WRITE. В UniPascal това не е разрешено.
- В ISO Pascal е дефинирана стандартната процедура PAGE, която предизвиква преминаване на нова страница. В UniPascal тази процедура не е дефинирана. Обикновено се използва WRITE(F,CHR(12)).
- В ISO Pascal са дефинирани съвместяеми масиви параметри. В UniPascal този вид параметри не са реализирани.
- В ISO Pascal е дефинирана стандартната константа MAXINT. В UniPascal тази константа не е дефинирана. Вместо нея може да се използва стандартната функция MAX(INTEGER).

F.2. Разширения спрямо ISO Pascal

- В UniPascal следните идентификатори са добавени към ключовите думи: **INTERFACE, IMPLEMENTATION, SEGMENT, UNIT, USES, XOR**.
- Символът за подчертаване '_' е равностоеен на буква.
- Целочислените константи могат да бъдат записвани в шестнадесетична система. За целта те трябва да бъдат предхождани от знака \$.
- Разрешена е употребата на идентификатори като етикети.
- Разширен е синтаксисът за низови константи. Те могат да съдържат контролни символи, съвместими са с дефинираните в UniPascal тип STRING, и могат да бъдат ограждани както с единични ('), така и с двойни (") кавички.
- Описанието на етикети, константи, типове, променливи, процедури и функции може да бъде в произволен порядък и разделите от всеки вид могат да се повтарят произволен брой пъти.
- Навсякъде, където в ISO Pascal се изисква константа, в UniPascal може да се използва константен израз.
- В UniPascal са реализирани допълнителни целочислени типове - SHORTINT, SHORTCARD, LONGINT, CARDINAL, NATURAL, а също така и допълнителните типове BYTE, WORD, LONGWORD.
- В UniPascal е реализиран стандартният тип string, който се различава от пакетираните масиви от символи с това, че включва динамична дължина на низа.

- Съвместимостта на типовете е разширена, за да разреши съвместимост между типовете STRING, CHAR и PACKED ARRAY OF CHAR.
- Типът на израз или променлива може да бъде сменен (Type Cast).
- Дефинирана е логическата операция XOR - изключващо или.
- Разрешено е използването на побитовите операции AND, OR, XOR над типовете BYTE, WORD и LONGWORD.
- Операциите за сравнение могат да бъдат използвани над пакетирани масиви от символи и над низове.
- Операциите за сравнение за равенство (=) и различие (<>) могат да бъдат използвани с всеки тип освен файловия.
- Синтаксисът на CASE оператора е разширен и включва ELSE част.
- Разширява се класификацията на параметри чрез въвеждане на нов вид - параметри-константи.
- Константните параметри и параметрите-променливи могат да бъдат нетипизирани, като съответните им фактически параметри могат да бъдат от произволен тип.
- Фактическите параметри, съответстващи на формални параметрите от тип BYTE, WORD и LONGWORD могат да бъдат променливи от всеки тип с големина съответно 1, 2 и 4 байта.
- В UniPascal са реализирани модули (UNIT), с което се осигурява реализацията на разделно-модулна компилация.
- В UniPascal са дефинирани следните стандартни процедури и функции, които не са дефинирани в ISO Pascal:

ADDR	HALT	MOVE
APPEND	HI	MOVEWORDS
BLOCKREAD	HIWORD	MSGLIMIT
BLOCKWRITE	INC	OPEN
CLOSE	INCL	PARAMCOUNT
DEC	INSERT	PARAMSTR
DELETE	INT	POS
EXCL	IORESULT	RANDOM
EXIT	LENGTH	RANDOMIZE
FILEPOS	LO	RELEASE
FILESIZE	LONGSWAP	RETURN
FILLCHAR	LOWORD	SEEK
FILLWORD	MARK	SIZEOF
FRAC	MAX	SWAP
FREEMEMWORD	MAXAVAIL	TRUNCATE
GETDATE	MEMAVAIL	UPCASE
GETMEMWORDS	MIN	SIZEOF
GETTIME		

- В UniPascal е възможно използването на четири възможни синтактични описания на операторите, с което четирите негови подмножества представляват съответно: езика Pascal; едно разширение на езика Pascal; език, подобен на Modula-2; едно разширение на Modula-2 подобният език.

