

1. Projekt oks-06

Základní informace:

■ Účel:

- naučit se vytvářet testovací scénáře
- prakticky se seznámit s nástrojem Squash
- naučit se připravit ze specifikace sady testovacích požadavků a od nich vytvářet testovací případy
 - ♦ a současně si uvědomit, že příprava funkčního testování může začít dříve, než vznikne/existuje testovaná aplikace
 - ta v tomto případě již existuje, ale pro návrh testů by stačila jen specifikace, případně i návrh či funkční prototyp GUI
- tam, kde je to vhodné, použít u funkčních testů rozdělení tříd ekvivalence
- **POZOR:**

Tento projekt vám možná bude připadat jako: “velmi zdlouhavý, nudný, buzerující, otrokářský, ...” (jsou použity termíny ze studentského hodnocení).

Jedním z cílů je, abyste si vlastnoručně vyzkoušeli a tím si jasně uvědomili, jak velké množství testů je potřeba pro řádné otestování i jednoduché aplikace. Dále si vyzkoušeli, že příprava testovacích scénářů není “jen otázka několika kliknutí”, které se provedou během pár minut, ale zodpovědná práce vyžadující čas, pečlivost a trpělivost.

V této souvislosti si též uvědomte, že máte předpřipravené kostry jak požadavků, tak testů, což velmi usnadňuje činnost, zvláště při existenci možnosti kopírování jednotlivých RQM i TC. Celkem se očekává, že podle vzorů vytvoříte 13 vlastních RQM a 50 vlastních TC.

■ **Kostra:** oks-data-06.zip

■ **Odevzdávaný soubor:** oks-06.jar

Zadání:

■ podle existující specifikace webové aplikace (viz níže) a jejího GUI (<http://oks.kiv.zcu.cz/OsobniCislo/Generovani>) doplnit v nástroji Squash již připravené kostry:

- skupin požadavků a požadavků
- testovacích sad a testovacích případů

Note

V tomto projektu se nic neprogramuje!

Popis vstupních dat:

■ stáhněte si soubor oks-data-06.zip a rozbalte jej

■ v adresáři `data` se nachází soubory

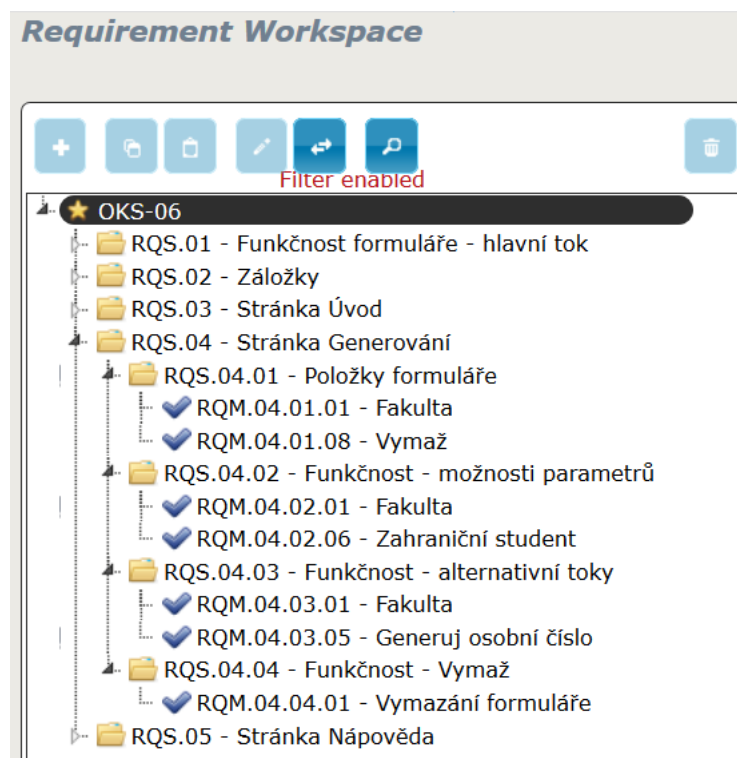
- `osobni-cislo-prototyp.html` - GUI prototyp stránky **Generování** z budoucí aplikace
 - ♦ počet, typy prvků a jejich rozložení budou v budoucí webové aplikaci stejné
 - ♦ bude pouze upraven jejich styl pomocí kaskádních stylů
 - ♦ tento soubor ve skutečnosti nepotřebujete, protože webová aplikace již fyzicky existuje
 - je zde jen pro ukázkou, že testovací scénáře lze napsat i jen podle prototypu či specifikace
- `OKS-06-kostra-RQM.json` - kostra požadavků
 - ♦ obsahuje všechny skupiny a podskupiny požadavků (RQS)
 - ♦ dále obsahuje ukázky jednotlivých požadavků (RQM) následovně (platí jen pro RQS.04):
 - pokud je ve skupině jen jeden nebo dva požadavky, jsou ponechány beze změny
 - pokud má být ve skupině tři a více požadavků, jsou ponechány pouze první a poslední z nich, např.:
 - skupina **RQS.04.01 Položky formuláře** má obsahovat ve finálním řešení celkem 8 požadavků, v předpřipravených datech obsahuje pouze:
 - **RQM.04.01.01 Fakulta** - první požadavek
 - **RQM.04.01.08 Vymaž** - poslední požadavek
- `OKS-06-kostra-TC.json` - kostra testovacích případů
 - ♦ obsahuje všechny testovací sady a jejich podskupiny (TS)
 - ♦ dále obsahuje ukázky jednotlivých testovacích případů (TC) následovně (platí jen pro TS.04):
 - pokud je v testovací sadě jen jeden nebo dva TC, jsou ponechány beze změny
 - pokud má být v testovací sadě tři a více TC, jsou ponechány pouze první a poslední z nich, např.:
 - testovací sada **TS.04.01 Položky formuláře** má obsahovat ve finálním řešení celkem 8 TC, v předpřipravených datech obsahuje pouze:
 - **TC.04.01.01 Fakulta** - první TC
 - **TC.04.01.08 Vymaž** - poslední TC
- `OsobniCislo-all-RQM.pdf` - ukázka kompletního řešení požadavků
 - ♦ tento soubor slouží jako **nápověda** v případě nutnosti
 - bylo by velkou chybou jej bezmyšlenovitě kopírovat

■ v adresáři `kontrola` se nachází soubor pro závěrečnou kontrolou

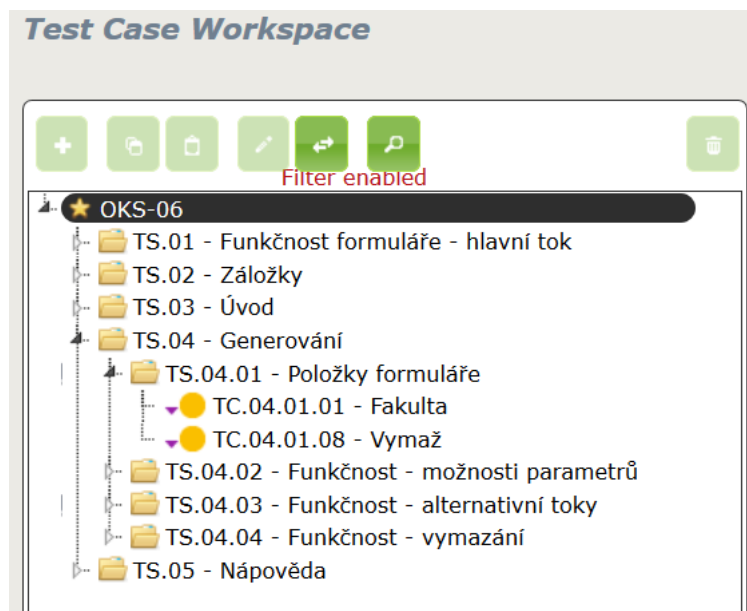
- `Kontrola_Prj_06.jar` - kontrolní program

Postup řešení:

- podle pokynů z přednášek si nainstalujte nástroj Squash - v tomto nástroji budou probíhat všechny aktivity
- spusťte Squash a přihlaste se jako administrator (`admin / admin`)
- pomocí aplikace GUI nebo CLI importu (viz přednášky) importujte do Squash soubory `OKS-06-kostra-RQM.json` a `OKS-06-kostra-TC.json` do projektu s nastavením:
 - OKS-06
 - Osobní číslo
 - OC
- výsledkem bude pro RQM



- a pro TC



- prohlédněte si jednotlivé skupiny požadavků (RQS) a jednotlivé požadavky (RQM) a zkuste je přesouvat, editovat, mazat apod.
 - v případě chyby lze kdykoliv znovu projekt OKS-06 zrušit, založit nově a znovu naimportovat oba soubory
- přečtěte si podrobně specifikaci uvedenou níže a prohlédněte si webovou aplikaci oks.kiv.zcu.cz/OsobniCislo/Generovani
- dále vás budou zajímat jen RQS.01 a RQS.04
 - RQS.01, RQS.02, RQS.03, RQS.04.04 a RQS.05 neměňte - jsou to již finální verze
- ve finální verzi RQS.01 se snažte pochopit způsob vytváření požadavků z existující specifikace

Warning

Je nezbytně nutné dodržet zavedený způsob identifikátorů!

- všechny další vaše doplnění budou jen v RQS.04.01, RQS.04.02 a RQS.04.03
 - doplňte zde všechny chybějící RQM
 - ♦ vždy si pozorně prohlédněte obsah ukázkových RQM v konkrétní skupině a v daném duchu doplňujte další RQM
 - ♦ velmi pečlivě vyplňte:
 - **Name** - např. Happy Day Scenario
 - **Reference** - např. RQM.01.01
 - ♦ u jednotlivých požadavků ponechte:
 - **Status**
 - **Criticality**
 - **Category**

- podobně postupujte pro TC
 - opět platí, že všechny TS jsou již kompletní a nebudete nijak doplňovat TS.01, TS.02, TS.03, TS.04.04 a TS.05
- stejným stylem jako u požadavků budete doplňovat chybějící TC do podskupin TS pouze v TS.04
 - zde se často s výhodou využije možnost kopírování již existujícího TC
 - velmi pečlivě vyplňte:
 - ◆ **Name** - např. Happy Day Scenario
 - ◆ **Reference** - např. TC.01.01
 - všimněte si, že pokud TC testuje víc věcí, pak je každá testovaná věc uvedena jednotlivě v samostatném kroku (viz TC.01.02)
 - ◆ to pak při vykonávání testu dává možnost snadno reportovat výsledek *pass* / *fail*
 - u jednotlivých TC ponechte:
 - ◆ **Status**
 - ◆ **Weight**
 - ◆ **Nature**
 - ◆ **Type**
 - nezapomeňte u každého TC provázat vytvářený TC s odpovídajícím RQM
- výsledné počty zjištěné z dashboardů celého projektu (zcela dole) by měly být:
 - RQM - 29
 - TC - 81
- po skončení celé práce exportujte pomocí aplikace GUI / CLI všechny požadavky do souborů OKS-06-RQM.json a OKS-06-TC.json
 - aplikace nabízí rozšířený název o datum exportu - po dokončení exportu jednoduše soubory přejmenujte

Kontrola úplnosti řešení:

- soubory OKS-06-RQM.json a OKS-06-TC.json překopírujte někam do svého pomocného adresáře, např. D:\zzz
 - dále sem překopírujte soubor Kontrola_Prj_06.jar z adresáře kontrola
- spusťte soubor Kontrola_Prj_06.jar


```
java -jar Kontrola_Prj_06.jar
```
- pokud je vypsáno hlášení typu:

Chybějící RQS nebo RQM: ...

a/nebo

Chybějící TS nebo TC: ...

- pak je příčina následující:
 - ♦ nejsou napsány všechny požadované RQM či TC
 - ♦ nejsou ve správném pořadí
 - ♦ nemají očekávané reference (identifikátory)

- pokud je vypsáno hlášení typu:

Za TC.01.01 není uvedeno pojmenování požadavku

- pak je příčina následující:
 - ♦ místo požadované dvojice reference a jména TC.01.01 Happy day scenario
 - ♦ byla použita jen reference TC.01.01

- v případě úspěšné kontroly nevypisuje kontrolní program žádné hlášení

Příprava souborů k odevzdání:

- v adresáři, ve kterém byla prováděna kontrola, použijte příkaz:

```
jar cMf oks-06.jar *.json
```

- výsledkem bude soubor oks-06.jar, který budete odevzdávat

- jeho velikost by měla být do 20 KB - pokud je větší, je v něm něco navíc

- po úspěšném odevzdání na Portále, proklikněte výsledné OK a, prosím, vyplňte v tabulce časovou náročnost této úlohy

1.1. Specifikace webové aplikace Osobní číslo

Note

Popisovaná aplikace je realizací požadavků na osobní číslo. Tyto požadavky jsou uvedeny níže.

1.1.1. Části aplikace

- aplikace se skládá ze tří stránek: Úvod, Generování a Návod
- navigace je řešena horizontálním menu umístěným v hlavičce

1.1.2. Stránka Úvod

- obsahuje stručné informace o aplikaci a odkazy na:

- webovou stránku s informacemi o předmětu KIV/OKS (na univerzitním Portálu)
- web KIV
- web ZČU

1.1.3. Stránka Náповěda

- obsahuje stručnou nápovědu k použití aplikace

1.1.4. Stránka Generování

- obsahuje formulář pro zadání údajů, ze kterých se následně generuje (skládá) výsledné osobní číslo
- všechny prvky (včetně labelů) mají své nápovědy (*tool-tip*)
- formulář má následující vstupní položky:

1.1.4.1. Fakulta

- povinná položka
- *tool-tip*: **Vyberte fakultu**
- realizována pomocí výběrového seznamu s položkami:
 - --- **neuveđeno** ---
 - **FAV - Fakulta aplikovaných věd**
 - **FDU - Fakulta designu a umění Ladislava Sutnara**
 - **FEK - Fakulta ekonomická**
 - **FEL - Fakulta elektrotechnická**
 - **FF - Fakulta filosofická**
 - **FPE - Fakulta pedagogická**
 - **FPR - Fakulta právnická**
 - **FST - Fakulta strojní**
 - **FZS - Fakulta zdravotnických studií**
- položka --- **neuveđeno** --- je předvybrána
- pokud není po stisku tlačítka **Generuj osobní číslo** vybrána některá z nabízených fakult:
 - výběrový seznam bude červeně orámován
 - vpravo se v červeném rámečku objeví **Fakulta nezvolena**

1.1.4.2. Rok nástupu na ZČU

- povinná položka
- *tool-tip*: **Zadejte poslední dvojčíslí kalendářního roku**
- realizována pomocí vstupního pole
 - jsou ignorovány bílé znaky na začátku a na konci zadaného řetězce
 - ◆ tyto bílé znaky jsou po odeslání formuláře ze zadaného řetězce bez varování odstraněny
- poslední dvojčíslí roku nástupu na studia
 - vždy dvě číslice včetně případné první nuly
 - ◆ nevýznamová (první) nula je povinná
 - hodnota **00** je povolena a udává rok 2000
- pole má zobrazenou nápovědu formátu pomocí hodnoty **08**
 - tato hodnota **není** předvyplnění
- pokud po stisku tlačítka **Generuj osobní číslo**:
 - není zadána žádná hodnota
 - ◆ bude vstupní pole orámováno červeně a vpravo se v červeném rámečku objeví **Prázdný vstup**
 - jsou zadány pouze bílé znaky
 - ◆ bude vstupní pole orámováno červeně a vpravo se v červeném rámečku objeví **Prázdný vstup**
 - je zadána pouze jedna dekadická číslice
 - ◆ bude vstupní pole orámováno červeně, zadané číslo v něm zůstane a vpravo se v červeném rámečku objeví **Je požadováno dvojčíslí**
 - jsou zadány více než dvě dekadické číslice
 - ◆ bude vstupní pole orámováno červeně, zadané číslo v něm zůstane a vpravo se v červeném rámečku objeví **Je požadováno dvojčíslí**
 - je zadána nečíselná hodnota
 - ◆ bude vstupní pole orámováno červeně, zadaný řetězec v něm zůstane a vpravo se v červeném rámečku objeví **Není dekadické číslo**
 - je zadáno znaménko (+1 nebo -1)
 - ◆ bude vstupní pole orámováno červeně, zadaný řetězec v něm zůstane a vpravo se v červeném rámečku objeví **Znaménko není povoleno**
 - je použita desetinná tečka nebo čárka (.1 1. ,1 1,)

- ♦ bude vstupní pole orámováno červeně, zadaný řetězec v něm zůstane a vpravo se v červeném rámečku objeví **Reálné číslo není povoleno**
- je použita vědecká notace v libovolné formě (1e2, 1E2, 1e-2, 1E-2)
- ♦ bude vstupní pole orámováno červeně, zadaný řetězec v něm zůstane a vpravo se v červeném rámečku objeví **Nepovolený zápis čísla**

1.1.4.3. Typ studia

- povinná položka
- *tool-tip*: **Vyberte typ studia**
- realizována pomocí výběrového seznamu s položkami:
 - **bakalářský**
 - **navazující**
 - **doktorský**
 - **magisterský**
- položka **bakalářský** je předvybrána
- v této položce není možné chybné nastavení - po stisku tlačítka **Generuj osobní číslo** zde nemůže být detekována (a následně vypsána) chyba

1.1.4.4. Pořadové číslo

- povinná položka
- *tool-tip*: **Zadejte nezáporné celé číslo, max. 4 číslice**
- realizována pomocí vstupního pole
 - jsou ignorovány bílé znaky na začátku a na konci zadaného řetězce
 - ♦ tyto bílé znaky jsou po odeslání formuláře ze zadaného řetězce bez varování odstraněny
- nezáporné čtyřmístné číslo udávající pořadí studenta podle abecedy v ročníku
 - vždy jedna až čtyři číslice včetně případných nevýznamových nul
 - ♦ nevýznamové nuly jsou nepovinné
 - hodnota prvního pořadí je **0001**
 - pořadí s hodnotou 0 není povoleno
- pole má zobrazenou nápovědu formátu pomocí hodnoty **0001**
 - tato hodnota **není** předvyplnění
- pokud po stisku tlačítka **Generuj osobní číslo**:

- není zadána žádná hodnota
 - ♦ bude vstupní pole orámováno červeně a vpravo se v červeném rámečku objeví **Prázdný vstup**
- jsou zadány pouze bílé znaky
 - ♦ bude vstupní pole orámováno červeně a vpravo se v červeném rámečku objeví **Prázdný vstup**
- je zadána nula jednou až čtyřmi číslicemi (0, 00, 000, 0000)
 - ♦ bude vstupní pole orámováno červeně, zadaný řetězec v něm zůstane a vpravo se v červeném rámečku objeví **Nula není povolena**
- jsou zadány více než čtyři dekadické číslice
 - ♦ bude vstupní pole orámováno červeně, zadané číslo v něm zůstane a vpravo se v červeném rámečku objeví **Více než čtyři číslice**
- je zadána nečíselná hodnota
 - ♦ bude vstupní pole orámováno červeně, zadaný řetězec v něm zůstane a vpravo se v červeném rámečku objeví **Není dekadické číslo**
- je zadáno znaménko (+1 nebo -1)
 - ♦ bude vstupní pole orámováno červeně, zadaný řetězec v něm zůstane a vpravo se v červeném rámečku objeví **Znaménko není povoleno**
- je použita desetinná tečka nebo čárka (.1 1. ,1 1,)
 - ♦ bude vstupní pole orámováno červeně, zadaný řetězec v něm zůstane a vpravo se v červeném rámečku objeví **Reálné číslo není povoleno**
- je použita vědecká notace v libovolné formě (1e2, 1E2, 1e-2, 1E-2)
 - ♦ bude vstupní pole orámováno červeně, zadaný řetězec v něm zůstane a vpravo se v červeném rámečku objeví **Nepovolený zápis čísla**

1.1.4.5. Forma studia

- povinná položka
- *tool-tip*: **Vyberte formu studia**
- realizována pomocí bloku radio-buttonů s hodnotami:
 - **prezenční**
 - **kombinovaná**
 - **distanční**
- není předvybrána žádná položka
- pokud není po stisku tlačítka **Generuj osobní číslo** vybrána některá z nabízených forem studia
 - vpravo se v červeném rámečku objeví **Forma nezvolena**

1.1.4.6. Zahraniční student

- nepovinná položka
- *tool-tip*: **Zaškrtněte, jedná-li se o zahraničního studenta**
- realizována pomocí check-boxu
- je-li položka vybrána, bude na konci výsledného osobního čísla písmeno **I** jako označení mezinárodních (*international*) studentů
- po spuštění aplikace není položka vybrána
- v této položce není možné chybné nastavení - po stisku tlačítka **Generuj osobní číslo** zde nemůže být detekována (a následně vypsána) chyba

1.1.4.7. Generuj osobní číslo

- realizováno pomocí tlačítka
- *tool-tip*: **Stiskněte po zadání všech hodnot**
- odešle všechny zadané hodnoty na server, kde jsou zkontrolovány

1.1.4.8. Vymaž formulář

- realizováno pomocí tlačítka
- *tool-tip*: **Vymazání všech hodnot**
- pošle zprávu serveru

1.1.4.9. Funkčnost formuláře

- po stisku tlačítka **Generuj osobní číslo**
 - v případě:
 - ◆ správných hodnot:
 - vypíše dolů do formuláře label **Výsledné osobní číslo** následovaný vygenerovaným osobním číslem
 - jednotlivé části vygenerovaného osobního čísla jsou popsány níže v sekci 1.5.
 - ◆ nesprávných hodnot:
 - vypíše hlavní chybové hlášení
 - je-li jakákoliv vstupní položka chybně vyplněna, vypíše pod formulářem červeně:
Chybné vstupy - opravte prosím zvýrazněné položky
 - každá konkrétní vstupní položka s nevalidními hodnotami je zvýrazněna a je u ní vypsán podrobnější popis chyby

- způsob zvýraznění a jednotlivá detailní chybová hlášení jsou popsána u každé položky - viz výše

♦ nevyplněného jakéhokoliv povinného pole:

– toto se považuje za jeden z případů výše popsaných nesprávných hodnot

■ po stisku tlačítka **Vymaž**

- vymaže obě vstupní pole a nastaví jim zobrazené nápovědy
- oba výběrové seznamy nastaví na předvyplněné hodnoty
- v bloku radio-buttonů nebude vybrána žádná hodnota
- check-box nebude vybrán
- všechna případná chybová hlášení vypsaná při předchozím generování budou vymazána
- bude vymazáno i případné výsledné vygenerované osobní číslo včetně svého labelu

1.1.5. Osobní číslo - možnosti a způsoby konstrukce

Note

Ve skutečnosti má konstrukce osobního čísla více možností, než je dále uváděno. Ale tyto možnosti jsou buď zastaralé nebo jsou využívány minoritně a my je nebudeme brát v úvahu.

1.1.5.1. Povolené znaky

■ jako znaky jsou povoleny jen

- dekadické číslice 0, 1, 2, ..., 9
- vybraná velká písmena anglické abecedy - viz dále části osobního čísla

1.1.5.2. Části osobního čísla

■ osobní číslo na ZČU se sestává z pěti (šesti) částí - `FrrTppppfN`, např. `A12B0987P`, kde:

1.1.5.2.1. **F** = Fakulta

■ jedno písmeno, které určuje fakultu

■ povoleny jsou znaky:

- **A** = FAV - Fakulta aplikovaných věd
- **D** = FDU - Fakulta designu a umění Ladislava Sutnara
- **K** = FEK - Fakulta ekonomická
- **E** = FEL - Fakulta elektrotechnická
- **F** = FF - Fakulta filosofická

- P = FPE - Fakulta pedagogická
- R = FPR - Fakulta právnická
- S = FST - Fakulta strojní
- Z = FZS - Fakulta zdravotnických studií

1.1.5.2.2. **rr** = rok nástupu na studia

- poslední dvojčíslí roku nástupu na studia
 - vždy dvě číslice včetně případné první nuly
- hodnota 00 je povolena a udává rok 2000

1.1.5.2.3. **τ** = typ studia

- jedno písmeno, které určuje typ studia
- povoleny jsou znaky
 - B = bakalářský
 - N = navazující
 - P = doktorský
 - M = magisterský (jen některé fakulty, např. FPR)

1.1.5.2.4. **pppp** = pořadové číslo

- pořadové číslo studenta
 - studenti jsou v prvním kole přijímacího řízení řazeni podle abecedy, takže pořadové číslo většinou odpovídá i pořadí jejich příjmení
 - ♦ může se stát, že student, který nastoupil na fakultu později, má vyšší pořadové číslo, než by odpovídalo abecednímu pořadí jeho příjmení a jména
 - ♦ nelze tedy činit žádné závěry ohledně příjmení studenta vyplývající z jeho pořadového čísla
- je vždy čtyřmístné s nevýznamovými nulami, začíná od 0001
 - číslování nemusí být spojitě
- mohou se mezi sebou míchat studenti různé formy studia (např. prezenční a kombinovaná)
- studenti různých typů (např. bakalářský a navazující) studia mají vždy svoji skupinu očíslovanou od 0001

1.1.5.2.5. **ε** = forma studia

- jedno písmeno, které určuje formu studia
- povoleny jsou znaky

- P = prezenční
- K = kombinovaná
- D = distanční

1.1.5.2.6. N = nepovinné

- jedno nebo více písmen, které určují doplňující informace
- nejčastěji se používá písmeno I jako označení mezinárodních (*international*) studentů, např. A14B0001PI